2 mf

DRA

A Reproduced Copy

OF

(NASA-TM-X-68288) NUMERICAL STUDIES OF UNSTEADY TWO DIMENSIONAL SUBSONIC FLOWS USING THE ICE METHOD Ph.D. Thesis - Toledo Univ. (NASA) 246 p HC $14.50

N73-31240

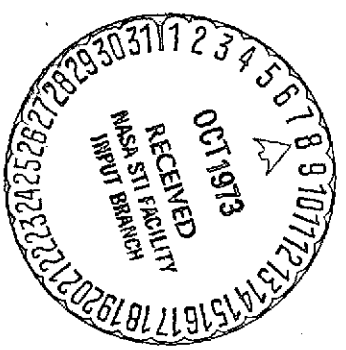
Unclas

CSCL 20D G3/12 14038

Reproduced for NASA

by the

NASA Scientific and Technical Information Facility



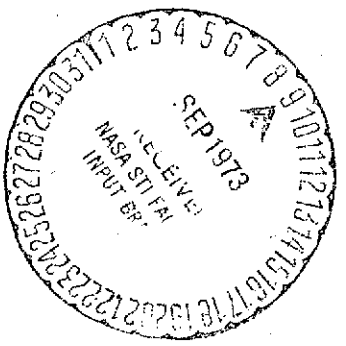
NASA TECHNICAL MEMORANDUM

NASA TM X-68288

NASA TM X-68288

NUMERICAL STUDIES OF UNSTEADY TWO-DIMENSIONAL SUBSONIC FLOWS USING THE ICE METHOD

by Paul R. Wieber
Lewis Research Center
Cleveland, Ohio 44135
August 1973



An Abstract of

NUMERICAL STUDIES OF UNSTEADY TWO-DIMENSIONAL SUBSONIC FLOWS USING THE ICE METHOD

Paul R. Wieber

Submitted in partial fulfillment
of the requirements of the
Doctor of Philosophy Degree
in Engineering Science

The University of Toledo
August 1973

A numerical program was developed to compute transient compressible and incompressible laminar flows in two dimensions with multicomponent mixing and chemical reaction. The algorithm used the Los Alamos Scientific Laboratory ICE (Implicit Continuous-Fluid Eulerian) method as its base. The program can compute both high and low speed compressible flows.

Point by point corrections were included for the errors caused by truncating the Taylor series during finite differencing. The removal of these numerical diffusion errors stabilized computations which previously had diverged catastrophically.

Multicomponent mixing and chemical reaction were incorporated using an implicit scheme and breaking the time increment into smaller steps. The mixing was computed over a small time step and was followed by reaction computations over a series of even smaller time steps. This effectively coupled the species equations and gave stable results.

The applicability of the computer program was tested with a variety of flow problems in tubes. These included flow startup in an infinite tube, shock tube flow, cyclical pulsations on a mean flow, uniform entry, coaxial entry into long and short tubes, flow of a center jet entering a sudden expansion, and steady parabolic coaxial entry with mixing and chemical reaction of trace species. The program was not proven for problems with strongly coupled flow and reaction. A variety of computer-drawn graphical output was used to display the results.

The numerical program incorporating the stabilization techniques was quite successful in treating both old and new problems. Detailed calculations of coaxial flow very close to the entry plane were possible. The program treated complex flows such as the formation and downstream growth of a recirculation cell. An implicit solution of the species equation predicted mixing and reaction rates which compared favorably with the literature. A recommendation for expanding the program capability to include strongly coupled flow and reaction was made. A computer program listing was furnished.

A Thesis

entitled

Numerical Studies of Unsteady Two-dimensional Subsonic Flows Using the ICE Method

by
Paul R. Wieber

as partial fulfillment of the requirements of
the Doctor of Philosophy Degree in
Engineering Science

_____
Adviser

_____
Dean of the Graduate School

The University of Toledo
July 1973

Dedication

To my very patient wife and family

Acknowledgements

The author wishes to express his appreciation to the following persons and organizations:

Dr. K. J. DeWitt who as thesis advisor gave technical guidance and inspiration, moral support, and friendship to the author during execution of the research;

Mr. Frank E. Belles of the National Aeronautics and Space Administration (NASA) for serving as co-advisor and providing many helpful and encouraging suggestions;

Mr. Sanford Gordon and Mr. Robert R. Hibbard (NASA) for their continued support and interest;

Dr. Frank J. Zelenznick (NASA) for shedding some light into the dimly lit corridors of partial differential equations;

Dr. David A. Bittker (NASA) for running his detailed kinetic program used for inputs into a reacting gas problem;

Mr. Roger A. Svehla and Mrs. Bonnie J. McBride (NASA) for providing thermodynamic and transport property data;

Mr. Alfred L. Armstead (NASA) for his invaluable teaching of the mysteries of time-sharing computers;

Dr. Leo F. Donovan, Dr. John Zuk, of NASA and Prof. John D. Hansell of the University of Toledo for many informative and spirited discussions concerning the computer program;

Dr. Francis H. Harlow and Dr. F. Daniel Butler of the Los Alamos Laboratory for their most valuable suggestions and information concerning the ICE technique.

The research was performed at the NASA Lewis Research Center. The author is especially indebted to NASA for the continued support which made this work possible.

TABLE OF CONTENTS

	Page
ACKNOWLEDGEMENTS	iii
LIST OF TABLES	vii
LIST OF FIGURES	vii
LIST OF APPENDICES	xi
NOMENCLATURE	xii
ABSTRACT	xvii
1. INTRODUCTION	1
2. LITERATURE BACKGROUND AND CONCEPTS	7
A. Concepts of Stability and Accuracy	8
B. Compressible Flow Schemes	12
C. The PIC, MAC, and ICE Methods	14
3. FORMULATION OF THE PARTIAL DIFFERENTIAL EQUATIONS	23
A. Tensor Forms	23
B. Dimensional Component Equations	27
C. Dimensionless Component Equations	28
4. FORMULATION OF THE FINITE DIFFERENCE EQUATIONS	34
A. The ICE Cell and Computational Mesh	34
B. Development of the Poisson Equation for Pressure	37
C. Final Finite Difference Forms	42
D. Boundary Conditions	53
E. Computer Solution of the Equations	58
5. STABILITY AND TRUNCATION ERROR ANALYSIS	61
A. Basic Concepts	61
B. Linear Stability Analyses	63
a) Method of Positive Coefficients	63
b) Method of Discrete Perturbation	64
c) Karplus' Method of Electric Circuit Stability	66
d) Method of Von Neumann	68
e) Method of Hirt	69
f) Application of the Various Methods	71
C. The Use of Hirt's Truncation Error Analysis	73
D. The Two-dimensional ICE Truncation Errors	77

TABLE OF CONTENTS (Continued)

		Page
6.	NUMERICAL RESULTS AND DISCUSSION.	83
	A. Introductory Comments	83
	B. Computer Facilities and Computer-Drawn Plots.	85
	C. Startup of Incompressible Laminar Flow in an Infinite Tube.	86
	D. The Shock Tube Problem.	88
	E. Incompressible Tube Flow with an Oscillatory Upstream Pressure Perturbation	91
	F. Transient Incompressible Entry Tube Flow.	94
	G. Stabilization of Coaxial Flow Calculations by the β Truncation Error Corrections.	97
	H. Short Tube with Coaxial Entry - Velocity Ratios = 2 and .05, Velocities = 10 and 20 cm/sec.	98
	I. Long Tube with Coaxial Entry - Velocity Ratios = 2 and .05, Velocities = 10 and 20 cm/sec.	102
	J. Center Jet Flow into a Larger Tube - Velocity Ratio = ∞, Velocity = 20 cm/sec	103
	K. Coaxial Parabolic Entry Flow with Specie Diffusion and Chemical Reaction	105
	L. Attempted Solution of Coupled Reaction and Fluid Mechanics	108
7.	CONCLUSIONS AND RECOMMENDATIONS	111
BIBLIOGRAPHY.		222

LIST OF TABLES

Table		Page
I	Description of Computations.	113

LIST OF FIGURES

Figure		Page
1	A Typical ICE Cell Showing the Variable Locations.	117
2	ICE Computational Grid	117
3	Basic Flow Sheet for the ICE Algorithm	118
4	Startup of Incompressible Flow in an Infinite Tube. Re = 205 at Steady State	119
5a	Pressure Profile for the Shock Tube Problem. t = .5. . . .	120
5b	Density Profile for the Shock Tube Problem. t = .5	120
5c	Velocity Profile for the Shock Tube Problem. t = .5. . . .	120
5d	Internal Energy Profile for the Shock Tube Problem. t = .5	120
6	Pressure Profiles for the Shock Tube Problem with Two Time Step Sizes. t = .5.	121
7	Pressure Profiles for the Shock Tube Problem using I^{n+1} in the Hybrid Function. t = .5	121
8	Effect of Time Centering on Pressure Profiles for the Shock Tube Problem. t = .125 and .5.	122
9	Effect of the Mass Truncation Error Correction, β_M, on a Density Profile for the Shock Tube Problem. The Calculation is Time-Centered, $\theta = \varphi = .5$.	122
10	Radial Distributions of the Axial Velocity Perturbations for Oscillatory Incompressible Tube Flow. $\sqrt{2\pi f/\nu}\ R_W = 3$. .	123
11	Radial Distributions of the Axial Velocity Perturbations for Oscillatory Incompressible Tube Flow. $\sqrt{2\pi f/\nu}\ R_W = 10$.	123
12	Radial Distributions of Axial Velocity Perturbations for Startup of Oscillatory Incompressible Tube Flow. $\sqrt{2\pi f/\nu}\ R_W = 5$.	124

LIST OF FIGURES (Continued)

Figure		Page
13	Cyclic Radial Distributions of Axial Velocity Perturbations for Transient Oscillatory Incompressible Tube Flow	124
14	Axial Velocity Development at Several Radial Positions for Laminar Incompressible Tube Entry Flow. t = 2.	125
15	Radial Profiles of Axial Velocity at Several Downstream Positions for Laminar Incompressible Tube Entry Flow. t = 2.	125
16	Radial Profiles of Radial Velocity at Several Downstream Positions for Laminar Incompressible Tube Entry Flow. t = 2.	126
17	Axial Profiles of Axial Velocity During Startup of Laminar Incompressible Tube Entry Flow. Values > 1 for R = .05; Values < 1 for R = .95	126
18	Radial Profiles of Axial Velocity During Startup of Laminar Incompressible Tube Entry Flow. Z = 1.333	127
19	Growth of Instability of Axial Velocity for Coaxial Entry into A Short Tube. R = .025, Entry Velocity Ratio = 2.0.	128
20	Stabilized Axial Velocity for Coaxial Entry into a Short Tube. R = .025, Entry Velocity Ratio = 2.0.	129
21	Plots of Transient Axial Velocity for Coaxial Entry into a Short Tube. Entry Velocity Ratio = 2.0 a) t = .002 b) t = .04 c) t = .10 d) t = .24.	130
22	Plots of the Transient Pressure Field for Coaxial Entry into a Short Tube. Entry Velocity Ratio = 2.0 a) t = .004 b) t = .24.	132
23	Coaxial Entry into a Short Tube. t = .24 (Steady State), Entry Velocity Ratio = 2.0 a) Radial Profiles of Axial Velocity b) Radial Profiles of Radial Velocities.	133

LIST OF FIGURES (Continued)

Figure		Page
24	Plots of Transient Axial Velocity for Coaxial Entry into a Short Tube. Entry Velocity Ratio = .05	
	a) t = .002 b) t = .05	134
	c) t = .10 d) t = .26	135
25	Coaxial Entry into a Short Tube. t = .26 (Steady State), Entry Velocity Ratio = .05	
	a) Radial Profiles of Axial Velocity	136
	b) Radial Profiles of Radial Velocity.	136
	c) Radial Profiles of Pressure	137
26	Plots of Transient Axial Velocity for Coaxial Entry into a Long Tube. Entrance Velocity = 2.0	
	a) t = .1 b) t = .5	138
	c) t = 1.0 d) t = 3.5	139
27	Coaxial Entry into a Long Tube. t = 3.5 (Steady State), Entry Velocity Ratio = 2.0	
	a) Radial Profiles of Axial Velocity	140
	b) Radial Profiles of Radial Velocity.	140
	c) Radial Profiles of Pressure	141
28	Coaxial Entry into a Long Tube. t = 2.4 (Steady State), Entry Velocity Ratio = .05	
	a) Radial Profiles of Axial Velocity	142
	b) Radial Profiles of Radial Velocity.	142
	c) Radial Profiles of Pressure	143
29	Transient Velocity Vector and Flow Direction Figures for Coaxial Entry, Center Jet Only. Velocity = 20 cm/sec	
	a) Velocity Field at t = .04	144
	b) Flow Direction at t = .04	144
	c) Velocity Field at t = .1	145
	d) Flow Direction at t = .1	145
	e) Velocity Field at t = .5	146
	f) Flow Direction at t = .5	146
	g) Velocity Field at t = 2	147
	h) Flow Direction at t = 2	147
30	Coaxial Entry, Center Jet Only. t = 2.0 (Steady State)	
	a) Radial Profiles of Axial Velocity	148
	b) Radial Profiles of Radial Velocity.	148
31	Coaxial Parabolic Entry into a Long Tube with Diffusion and Reaction of Trace Species	
	a) Radial Profiles of Axial Velocity	149
	b) Radial Profiles of Radial Velocity.	149

LIST OF FIGURES (Continued)

Figure		Page
32	Comparison with Seider's No-Reaction Computation. $Z = 12$, $N_{Re} = 248$, $N_{Sc} = .942$.	150
33	Parabolic Coaxial Entry into a Long Tube with Diffusion and Reaction of Trace Species	
	a) Radial Profiles of Hydrogen Concentration	151
	b) Radial Profiles of Iodine Concentration	151
	c) Radial Profiles of Hydrogen Iodide Concentration. .	151
34	Contour Plot of Hydrogen Iodide Concentration. Parabolic Coaxial Entry into a Tube 20 Radii Long.	152
35	Parabolic Coaxial Entry into a Tube 5 Radii Long with Diffusion and Reaction of Trace Species	
	a) Radial Profiles of Axial Velocity	153
	b) Radial Profiles of Hydrogen Iodide Concentration. .	153
36	Contour Plot of Hydrogen Iodide Concentration. Parabolic Coaxial Entry into a Tube 5 Radii Long	154
37	Onset of Instability in Temperature Profiles for the CO Oxidation Problem	155
38	Number of Iterations Required to Converge the Pressure Field vs. and ADI Scaling Parameter.	159
39	Curve Fit of Detail Kinetic Calculations to Give a Global Rate. Second Order Oxidation of Carbon Monoxide. .	163

LIST OF APPENDICES

Appendix		Page
A	INVERSION OF THE $\tilde{P}$ MATRIX.	156
B	OUTLINE OF THE ALGORITHM FOR FLOWS WITH STRONG REACTION	160
C	DESCRIPTION OF THE NUMERICAL PROGRAM	164
	A. Programs	165
	B. Equation Variables, Constants and Supporting Information.	165
	C. Program Control Variables.	168
	D. Listing of Computer Program.	169

NOMENCLATURE

Roman Letters

A	speed of sound, squared
a	speed of sound
B	explicit collection of terms, equation (4-23)
C	concentration
C_P	heat capacity at constant pressure
C_V	heat capacity at constant volume
D	explicit collection of terms, equation (4-24); also multi-component diffusion coefficient
$\mathscr{D}$	binary diffusion coefficient
E	total energy
e	exponential
F	fractional volume; also general term
f	frequency
G	explicit collection of terms, equation (4-25)
g	metric tensor
H	amplification factor; also enthalpy
I	internal energy
i	radial cell index; also current; also $\sqrt{-1}$
j	axial cell index
K	empirical constant in equation (2-5)
k	thermal conductivity
k'	reaction rate
M	molecular weight; also mass
m	mass

N_{Eu}	Euler number
N_{Pr}	Prandtl number
N_{Re}	Reynolds number
N_{Sc}	Schmidt number
$N_{S\ell}$	Strouhal number
n	mass flux; also cycle number
P	pressure
$\tilde{P}$	Hybrid function, equation (4-16)
Q	velocity divergence, equation (4-22)
q	heat flux; also artificial dissipation
R	radial position
R_A	aspect ratio
R_C	gas constant
R_W	tube outer radius, the reference length
$R_1 \rightarrow R_4$	resistances
r	radial position; also chemical generation term
S	collection of explicit terms from species equation; also Fourier coefficient
T	temperature
t	time
$t_\sim$	cycle time, the reference time
U	axial velocity
V	radial velocity; also, reference velocity
V_s	specific volume
v	vector velocity, when used with tensor notation
W	general term
X	mole fraction

Y	number of steps in which each diffusion time step is divided
y	the reaction time step number
Z	axial position
z	axial position

Greek Letters

ρ	density
τ	stress tensor; also phase angle
μ	viscosity
Δ	incremental step
ε	third order tensor; also, convergence error on the pressure iteration
γ	ratio of heat capacities
π	3.1416
φ	momentum equation proportioning coefficient
θ	mass equation proportioning coefficient
β	truncation error correction
α	relaxation coefficient
χ	pseudo time in ADI equation
ψ	species equation proportioning coefficient
ν	kinematic viscosity

Subscripts

eff	effective
i, j, k, m, n, p, q, s	tensor notation

i	radial cell index
j	axial cell index
K	Kth component or term
L	Lth component
M	mass; also component M
MK	species mass
R	Rth component
r	radial
R-K	mixture R which excludes K
tot	total
VR	radial momentum
VZ	axial momentum
Z	axial
z	axial

Superscripts

A	molar reaction coefficient for component K
B	molar reaction coefficient for component L
i, j, k, m, n, p, q, s	tensor notation
n	time index
Q	iteration index
*	reference condition; also intermediate value

Abbreviations

ADI	Alternating Direction Implicit
CPU	Central Processing Unit
FLIC	Fluid in Cell
ICE	Implicit Continuous - Fluid Eulerian
LASL	Los Alamos Scientific Laboratory
MAC	Marker and Cell
OPDE	Original Partial Differential Equation
PIC	Particle in Cell
SMAC	Simplified Marker and Cell
SOR	Successive Over Relaxation
TSS	Time Sharing System
3-D	Three Dimensional

Miscellaneous

$\overline{}$	dimensional (overbar)
~	temporary (overbar)

ABSTRACT

A numerical program was developed to compute transient compressible and incompressible laminar flows in two dimensions with multicomponent mixing and chemical reaction. The algorithm used the Los Alamos Scientific Laboratory ICE (Implicit Continuous-Fluid Eulerian) method as its base. The program can compute both high and low speed compressible flows.

Point by point corrections were included for the errors caused by truncating the Taylor series during finite differencing. The removal of these numerical diffusion errors stabilized computations which previously had diverged catastrophically.

Multicomponent mixing and chemical reaction were incorporated using an implicit scheme and breaking the time increment into smaller steps. The mixing was computed over a small time step and was followed by reaction computations over a series of even smaller time steps. This effectively coupled the species equations and gave stable results.

The applicability of the computer program was tested with a variety of flow problems in tubes. These included flow startup in an infinite tube, shock tube flow, cyclical pulsations on a mean flow, uniform entry, coaxial entry into long and short tubes, flow of a center jet entering a sudden expansion, and steady parabolic coaxial entry with mixing and chemical reaction of trace species. The program was not proven for problems with strongly coupled flow and reaction. A variety of computer-drawn graphical output was used to display the results.

The numerical program incorporating the stabilization techniques was quite successful in treating both old and new problems. Detailed calculations of coaxial flow very close to the entry plane were possible. The program treated complex flows such as the formation and downstream growth of a recirculation cell. An implicit solution of the species equation predicted mixing and reaction rates which compared favorably with the literature. A recommendation for expanding the program capability to include strongly coupled flow and reaction was made. A computer program listing was furnished.

Chapter 1

INTRODUCTION

The use of numerical methods to solve the partial differential equations of fluid flow is a field of great current interest. Advances in numerical techniques plus new generations of extremely fast digital computers permit solutions of complex flow problems. These solutions may be run as computer experiments, allowing the quantification of variables that would be difficult to measure in a physical experiment. The knowledge gained from such computer experiments can be used by scientists to advance the frontiers of fluids research. At the same time the expanded capability in solving the flow equations permits treatment of more complex and hence more realistic problems. Thus numerical fluid dynamics also has increasing value to the engineer.

Early numerical solutions dealt with steady laminar incompressible flows, steady compressible flows at supersonic speeds, and transient one-dimensional flows associated with shock waves. Many of these problems were selected because analytical solutions were available for comparison.

Then problems were run with more complex boundary conditions. An interest grew in obtaining transient solutions and three dimensional steady solutions. The current generation of very fast, large storage machines can treat some unsteady problems in three dimensions with

reasonable computation times. However, finding a means to display such solutions is a formidable problem in itself.

The mathematical problems of stability and convergence have also increased with problem complexity. The nonlinear nature of the differential equations precludes a rigorous analysis to characterize the validity of the numerical solution. Thus the researcher is often forced to identify a stable, "reasonable-looking" solution as a valid solution. This is not always true, but in the absence of rigorous proofs, it is a reasonable assumption.

Detailed computations of turbulent flows are presently unattainable. Turbulent flows consist of a myriad of transient, three dimensional fluctuations superimposed on a mean flow. Ignoring the transient nature of the fluctuations and reducing the dimensionality from three to two are already severe approximations to the actual physics of the flows. The turbulent behavior is dependent upon small scale motions. The extremely small mesh size, hence large numbers of grid points, required to resolve the small-scale turbulent motion demands enormous speed and storage capabilities.

Many flows of interest to engineers are in the turbulent region and these practical problems require solution. The need has been met by using semi-empirical models and curve fits of data to evaluate the turbulent coefficients. This process usually includes an order of magnitude analysis to cast out some stress components. A minimum grid size is chosen consistent with computer capabilities, and the effects of smaller scale flows are expressed as eddy viscosities. The method often works well for steady state problems where the model is valid and within a

parametric range embraced by the data. Extrapolation of such analyses to a wider operating range is often unsatisfactory. Transient problems involving the generation, propagation, and dispersion of turbulent quantities have been formulated, but little has been done in solving this type of problem, again due to formidable computational problems.

Solutions of the rigorous conservation equations are constrained entirely to the laminar region. A major current goal in numerical fluid dynamics is the solution of the complete laminar conservation equations for various sets of initial and boundary conditions. In addition, combustion and pollution problems have stimulated interest in reacting flows. And in some cases, such as acceleration of gaseous flows to high speeds or pulsatile flows, simplifying assumptions regarding compressibility can not be made. Gas phase reactions can cause energy release that couples with the fluid dynamics through compression effects. Hence a general numerical program to deal with these types of problems is of interest and has been pursued in this work.

A major problem in dealing with transient flows at all speeds is that the full conservation equations can exhibit the behavior of the three classes of partial differential equations, depending on the nature of the physical problem being described. Transient slow flows which are appreciably influenced by viscosity are parabolic in nature. Steady subsonic flows are described by elliptic equations. Subsonic flows with wave propagation and all supersonic flows are hyperbolic in nature. The difference equations which represent the differential equations generally require a method of solution that depends upon the class of equation. Thus a scheme that successfully solves hyperbolic equations may

fail in the elliptic region. Mixed flows of variable nature are difficult to solve numerically.

The conservation equations governing the flow of multi-specie reacting fluids are given below in tensor form. Gravity forces are neglected, g^{ji} is the metric tensor, and the comma (,) denotes covariant differentiation. Symbols are defined in the nomenclature.

$$\text{Mass:} \quad \frac{\partial \rho}{\partial t} = - (\rho v^j)_{,j} \tag{1.0-1}$$

$$\text{Momentum:} \quad \frac{\partial (\rho v^j)}{\partial t} = - (\rho v^j v^i)_{,i} - g^{ji} P_{,i} - \rho^{ji}{}_{,i} \tag{1.0-2}$$

$$\text{Total Energy:} \quad \frac{\partial (\rho E)}{\partial t} = - (\rho v^j E)_{,j} - (P v^j)_{,j} - q^j{}_{,j} - (\tau^{mn} g_{nj} v^j)_{,m} \tag{1.0-3}$$

$$\text{Mass of Kth Specie:} \quad \frac{\partial (\rho_K)}{\partial t} = - (n_K)^j{}_{,j} + (r_K) \tag{1.0-4}$$

The above equations all contain an accumulation term on the left hand side which is equated to various fluxes on the right hand side. The fluxes are convective or diffusive or arise from the action of pressure forces and viscous stresses. In the species equation a source term (r_K) is included to take chemical reaction into account.

For a compressible Newtonian fluid the viscous stress tensor is,

$$- \tau^{ji} = - \frac{2}{3} \mu g^{ji} v^m{}_{,m} + \mu g^{mj} g^{ni} (v_{m,n} + v_{n,m})$$

which may be substituted into the momentum and energy equations. These equations comprise the set to be solved for various initial and boundary conditions.

This thesis presents numerical studies of transient, two dimensional, confined flows with no constraints on fluid compressibility or

flow speed. The basic algorithm was derived at the Los Alamos Scientific Laboratory (LASL). The diffusion and reaction of species is included herein to broaden the range of application of the computer code.

The purposes of this thesis are:

(1) To develop the partial differential component equations through finite differencing and to formulate the algorithm;

(2) To derive and demonstrate a nonlinear truncation error correction to stabilize solutions;

(3) To demonstrate the wide applicability of the algorithm by comparing numerical solutions with other analytical and numerical solutions for the following problems:

(a) Startup of incompressible flow in an infinite tube;

(b) Propagation of shock phenomena after removing a diaphram separating high and low pressure gases;

(c) Development of incompressible flow from rest when the axial pressure gradient has a mean component and an oscillatory perturbed component;

(d) Development of the incompressible boundary layer in the entrance of a tube subjected to a uniform input flow;

(e) Calculation of steady-state flows with trace (decoupled) chemical reactions for fully developed coaxial entry.

(4) To present transient and steady state solutions of compressible and incompressible slow speed tube flows with uniform coaxial entry;

(5) To present transient and steady-state solutions of the incompressible and compressible flow of a slow speed center jet into a confined tube of larger diameter;

(6) To furnish a listing of a computer program for future experimentation with flows strongly coupled with chemical reaction.

Chapter 2

LITERATURE BACKGROUND AND CONCEPTS

A comprehensive review of publications on computational fluid mechanics would be a major undertaking and inappropriate to the intent of this thesis. This section will discuss some papers on concepts of stability and accuracy of difference schemes plus methods of solving compressible flows which can extend into the subsonic region. Some of the major techniques developed at the Los Alamos Scientific Laboratory to treat compressible and incompressible flows will be discussed, leading to the ICE technique for flows at all speeds.

All attempts to produce computer solutions which describe fluid flows are faced with the mathematical problems of existence, uniqueness, convergence, stability and accuracy of the solutions. The practical application of the numerical computation is to describe a physical problem. In the case of fluid flows, the use of the full conservation equations gives the greatest confidence that the physics of the problem are adequately represented. Since the conservation equations are a set of nonlinear partial differential equations, the mathematical problems mentioned above have not yet been resolved.

Roache (45) has considered this problem and concluded that physical intuition, heuristic reasoning, and numerical experimentation comprise a reasonable alternate approach. Since a goal of this work is

to develop and test a program applicable to physical problems of interest to engineers, the same philosophy is adopted here.

A. Concepts of Stability and Accuracy

The paper of Courant, Friedricks, and Lewy (12) provided the basis for constructing stable difference schemes. Elliptic, parabolic, and hyperbolic equations were treated. According to Lax (37), the authors were primarily interested in proving the existence of solutions to differential equations by taking solutions of finite difference equations to the limit of smallness. Their analysis of hyperbolic equations defined a "domain of dependence," a region of information on space and time coordinates which must be considered to define the value of a variable at a point. They showed that the domain of dependence of the difference equation must include the domain of dependence of the differential equation. Otherwise numerical instability will occur. In simpler terms, instability will arise if the grid spacing ΔZ is so large that information can not propagate from one point to the next in the given time increment Δt.

The above concept is expressed in the Courant number. The speed of propagation for compressible problems is the speed of sound, a. Then the Courant number restricts

$$a \frac{\Delta t}{\Delta Z} < 1$$

if stability is to be possible. In the case of incompressible flows, the formulation of the equations suppresses the sonic signals. The parameter propagating the information is the local fluid velocity, U. Then

$$U \frac{\Delta t}{\Delta Z} < 1$$

is the stability criterion.

The presence of shocks in flows presents computational problems using the inviscid equations because the primary variables are discontinuous over the shock front. This requires a set of internal boundary conditions obtained from the Rankine - Hugoniot equations to connect the variables over the shock front. The shock surface is usually positioned between grid points, and its exact position can be found only by a trial and error process. Von Neumann and Richtmeyer (51) introduced the concept of adding an artificial dissipation term to the equations. This caused the shock front to smear over a few grid points, eliminating the discontinuous front and the associated computational headaches. Shock strength and position automatically arose from the fluid flow calculations. A properly defined dissipative function produces negligible effects on the shock velocity and strength and no effects outside of the shock region. A real shock is slightly smeared due to heat conduction and viscosity effects. If shock spatial details are the primary goal of the computational exercise, the artificial dissipation term would interfere. For most shocked flows this is not of interest.

Von Neumann and Richtmeyer introduced the dissipative function, q, as an additive term on pressure in the momentum and energy equations. For unsteady one-dimensional flow, the equations were

$$\rho \frac{\partial U}{\partial t} = -\frac{\partial}{\partial Z} (P + q) \qquad (2\text{-}3)$$

$$\frac{\partial E}{\partial t} = - (P + q) \frac{\partial V_s}{\partial t} \qquad (2\text{-}4)$$

The dissipation term chosen was

$$q = - \frac{(K\Delta Z)^2}{V_s} \frac{\partial U}{\partial Z} \left| \frac{\partial U}{\partial Z} \right| \qquad (2\text{-}5)$$

This functional form produced a shock wave of thickness $O(\Delta Z)$, satisfied the Hugoniot relationships, and vanished away from the shocked region.

The theory of the stability of linear difference equations with constant coefficients was developed by Von Neumann at Los Alamos during World War II. The first detailed explanation of the method was published in the unclassified literature by O'Brien et al. (40). The authors illustrated Von Neumann's method of substituting an exponential series solution into the difference equation and examining the result for regions of time and space grid sizes which assured that the exponential terms would not grow with time. This theory will be discussed in more detail in the later section on stability analysis.

Cheng (11) examined computational stability, accuracy, and consistency of difference formulations. His discussion was based on Lax's equivalence theorem which states that, "In the limit $\Delta t \to 0$ and $\Delta Z \to 0$, the solution of the difference formulation converges to the solution of the differential problem if, and only if; (1) the differential problem is well-posed, (2) the difference and differential formulations are consistent, and (3) the computation is stable." Cheng analyzed the aspect of consistency in terms of the truncation errors inherent in the particular finite difference form. The truncation

error is a measure of disagreement between the partial differential equation and its finite difference equation. Cheng recognized the truncation error terms as dissipative, dispersive, and of higher order, then analyzed stability in a heuristic manner, looking at the sign of the dissipative term to determine whether a difference scheme was stable or unstable. A similar analysis by Hirt (29) is used in this thesis and is discussed later. Cheng calls the dissipative term a "pseudo-diffusivity." It is not Von Neumann's artificial viscosity which is an added term. It is implicitly contained in the finite difference formulation.

After an analysis of accuracy, Cheng concluded that viscosity controlled problems should have finite differences of second order accuracy which reduce the pseudo-diffusivity. Thus the true viscous effects would not be masked by numerical error. He also concluded that steady-state or slow transient solutions to the Navier-Stokes equations could be accurately computed. However, he showed that the second-order accurate Lax Wendroff method gave appreciable amplitude and phase errors when applied to a rapidly oscillating problem with known analytical solution. This specific example indicated that caution is required when solving rapidly changing time-dependent problems.

A more detailed linear analysis of amplitude and phase errors was conducted by Fromm (20). He examined the unsteady vorticity formulation of the incompressible flow equations by substituting a Fourier component solution and examining the stability of the high frequency components. He concluded that first order accurate formulations had appreciable amplitude and phase errors in the smaller modes, and that even second

order formulations had significant phase lag at the smallest mode, $2\Delta Z$. This caused a dispersion and appearance of point-to-point waves. Fromm proposed a fourth order scheme which showed superior error characteristics. He also proposed using a linear combination of two schemes with opposite phase error to cancel the error. This promises to be a fruitful area for future research.

B. Compressible Flow Schemes

Some of the schemes currently in use to treat transient compressible flows are briefly discussed in this section. Brailovskaya (5) presented an explicit differencing scheme for the laminar unsteady compressible Navier-Stokes equations which was first order accurate in time and second order accurate in the space variables. The scheme was three level. Intermediate values for mass, momentum, and energy fluxes were calculated explicitly, then advanced time values were computed using the intermediate values in the flux and pressure terms while reusing the old dissipation terms. A computation was shown for a cavity problem where a wall impulsively started moving across a square cavity, setting up a vortex within the cavity. Little detail was shown, but a steady-state streamline plot at a Reynolds number of 500 looked similar to other solutions of this problem. The lowest Mach number of the moving wall was 1/3. It was not stated whether the fluid was considered to be at rest on the walls.

Kurzrock and Mates (35) presented a method which they claimed had storage and computation speed advantages over Brailoyskaia's method. The continuity equation was written implicitly, but the new fluxes were

calculated explicity from the momentum equations and then used directly in continuity. Hence the scheme was computationally explicit. In fact, trial runs with an explicit form of the continuity equation gave the same stability limits as the implicit form. Evidently stability was controlled by the momentum and energy equations. The method was used to calculate a shock propagating inside a tube including boundary layer effects and for an external flow problem describing the growth of a shock on the leading edge of a flat plate suddenly accelerated to supersonic speed. An attempt to run the latter problem at subsonic velocity was thwarted by instability and long computation times.

A method by MacCormack (39) is applicable to compressible flows and has second order accuracy in both time and space. It is a three level method of the Lax-Wendroff type, in which an approximate set of values at $t + \Delta t$ is calculated from the starting values using spatial differences that step forward. The approximate values are then used in the second half of the scheme to calculate the true values at $t + \Delta t$ from differences that step backward. It has been used with supersonic and transonic flows. Although this writer has had personal communications with several researchers who indicated the method could be used for very slow flows, publications have not been found to verify the low speed behavior of the scheme.

The Lax-Wendroff scheme referred to in the previous paragraph deserves further mention. It is used with the inviscid equations rather than with the full Navier-Stokes equations. The one dimensional formulation was presented by Lax and Wendroff (38). Burstein (9) extended the scheme to two dimensions, but found that artificial viscosity was

needed to stabilize computations. His sample calculation of supersonic flow over a blunt body had a subsonic region which was handled successfully. The method is second order and has superior accuracy. It is a three level scheme requiring considerable averaging and is difficult to use.

Although the methods discussed in this section have varying success in the subsonic region, they are all constrained by the Courant condition. For one dimension, this is approximately

$$(|U| + a)\frac{\Delta t}{\Delta Z} < 1 \qquad (2\text{-}6)$$

Thus the time increment is constrained by the large value of the speed of sound even though the flow velocity may be very small.

C. The PIC, MAC, and ICE Methods

The Los Alamos Scientific Laboratory (LASL) is a pioneer in the field of numerical fluid dynamics. Much of the early work performed there was done in the support of weaponry for World War II. Open publications began to appear in the 1950's. This section introduces three of the LASL numerical fluid dynamic schemes, the PIC (Particle-in-Cell) method for compressible flows, the MAC (Marker-and-Cell) method for incompressible flows, and the ICE (Implicit Continuous-Fluid Eulerian) method for compressible flows at all speeds. The numerical program developed in this thesis is based on the ICE technique.

The PIC method was developed in the 1950's, but a later publication by Amsden (2) is used for this discussion. The PIC method was a hybrid scheme which permitted Lagrangian particles to flow through an Eulerian mesh, carrying with them the conservation properties of mass,

momentum, and energy. The computational fluid was thus not a continuum, but an assembly of particles. Multiple fluids could be denoted by different types of particles. The conservation, state, and thermodynamic properties of the contents of each cell were calculated using the number and types of particles lying within the cell boundaries. The number of particles in a cell varied due to flow and if the mean number was low, the cell properties oscillated in a bounded manner. The mass of each particle remained constant, so mass was automatically conserved by the method. It was thus not necessary to separately solve the mass equation.

The calculation took place in two phases. The first phase was Eulerian and presumed no particle movement. The steps were:

1) Compute the cell pressure from an equation of state

$$P^n = P\left(\frac{M}{F}, I\right) \tag{2-7}$$

where n denotes the time coordinate index.

2) Estimate a tentative new velocity $\tilde{U}$ from the equations of motion with advection terms removed. This step required an artificial viscosity for stabilization.

3) Estimate a tentative new internal energy $\tilde{I}$ neglecting advective terms in the energy equation. This ended the Eulerian phase.

The Lagrangian phase allowed particle movement and included effects of particles changing cells. The steps were:

1) Calculate the total momentum $(\tilde{\rho U})_{tot}$ of each cell containing N particles by

$$(\widetilde{\rho U})_{tot} = \tilde{U} \sum_{K=1}^{N} m_K \qquad (2\text{-}8)$$

where m_K was the mass of the Kth particle in the cell. Likewise calculate the cell internal and total energies.

2) Move the particles by assigning to each an effective velocity U_{eff} which was a weighted average of the surrounding velocities $\tilde{U}$: then translate the coordinate Z by

$$Z^{n+1} = Z^n + U_{eff}\Delta t \qquad (2\text{-}9)$$

3) If a particle crossed a boundary, adjust the cells for the changes in mass, momentum, and total energy. Then calculate the final velocities by dividing the adjusted momenta by the adjusted masses. Use the final velocities to calculate the final internal energies from the final total energies. This concluded the process.

The original concept of this method arose from particulate kinetic theory, the details having been developed over a long period of numerical trials. Besides the added artificial viscosity q, the method had an implicit dissipative term (Cheng's pseudo-diffusivity) which had the form $\frac{1}{2}\rho|U|\Delta Z$ in the Z direction. This term stabilized the computation. At low velocities the implicit damping term became small and instability resulted. In addition the Courant condition restricted the time increment size, so the method could not be used successfully for slow speed flows.

The method suffered from accuracy not only due to the first order differential approximations but because a continuum flow was represented by motion of a small number of particles. It did have the

capability of treating flows with more than one material and flows involving large fluid distortions. In addition, the particles showed the positions of the fluid, and their discrete translations could be filmed to pictorially show the fluid motion.

Amsden (2) presented calculations of high speed wakes, shock interaction with a blunt object, explosive burning, and high velocity jet splash and particle impacts. The pictorial solutions looked qualitatively correct. Evans and Harlow (16) calculated the flow over a cylinder in a channel using PIC. The cylinder was impulsively accelerated to steady Mach numbers of 2, 4, and 6. The streamlines and shock positions at steady state compared well with experimental values. Harlow and Meixner (28) calculated the rise of a hot gas bubble from the earth's surface, modeling a nuclear explosion in the atmosphere. The solution showed the shock front breaking away from the heated gas front and vertical stratification within the bubble due to shock-rarefaction interactions.

A natural progression of the PIC method was to eliminate the particles as carriers of conservative properties and to use a continuum. This was done by Rich (43) and later by Gentry et al. (21). They retained the same calculation sequence as PIC but solved the equations in integral form, relating the change of the property within the cell to fluxes through cell surfaces. The equation for conservation of mass was required in the second phase of the calculation since the mass-carrying particles were discarded. The absence of particles identifying the fluid made two-fluid computations difficult. This problem was treated by Rich by initially specifying the position of

the interface between two fluids, then tracking the interface as the problem progressed. Gentry's scheme, called FLIC for Fluid-in-Cell, optimized the computation for one fluid and prescribed a treatment for boundaries that required partial cells. Problems of supersonic flow over obstacles and diffraction of a shock travelling down a Z-shaped tunnel compared well with experimental results.

Concurrently with the PIC method for compressible flow, the MAC (Marker-and-Cell) method was developed for incompressible flow. A later summary document is Welch et al. (55). The differential form of the conservation equations were used in MAC instead of the integral form. A Poisson equation for pressure was formed by substituting the momentum equations into continuity. The Poisson equation was solved implicitly by relaxation to a specified degree of convergence to give the advanced pressure field. A technique developed by Hirt and Harlow (31) was used to correct for the lack of exact convergence. Pressure signals in incompressible flows are presumed to propagate instantly to all other points in the flow field. Thus the pressure at any point is a function of the entire velocity field, not just the local field. The implicit nature of the pressure equation served to carry the flow information throughout the pressure field.

After the iteration on the pressure field, the new velocities were calculated from the advanced time pressure term and the old shear terms. Then the particles were moved according to their weighted average velocities acting over the time increment Δt. The particle concept from the development of PIC carried over into the incompressible program, but in MAC, like FLIC, the particles served only as markers to identify the

position, type, and motion of the fluids. The presence of a free liquid-gas surface was included, and the primitive variables of velocity and pressure were retained to facilitate this capability. Later improvements in the MAC numerical code led to the simplified MAC or SMAC method (3).

A fascinating series of numerical problems were pictorially presented in Welch et al. (55), Harlow and Amsden (26), and Amsden and Harlow (3). These included a wave breaking on a sloping beach, a wall of water hitting an obstacle, water flow from a sluice gate, formation of a hydraulic jump, waterfalls, fountains, and others. Further application to multiple immiscible fluids was demonstrated by Daly (13) who studied Rayleigh-Taylor instability in a system with a dense fluid initially layered above a lighter fluid. Mushroom-shaped spikes of fluid were formed in the numerical calculation as the fluids began to invert to a stable configuration. These spikes were observed experimentally. Quantitative comparisons of growth of the protuberances were favorable. The MAC technique was extended by Hirt and Cook (30) to three dimensions including thermal buoyancy. Using 3344 cubic cells they computed flows over and around configurations of rectangular solids. Such calculations could be applied to meteorology and atmospheric pollution problems. They used the cell markers to follow the dispersion of a tracer from a point source. The results were displayed in perspective drawings.

The MAC method has been used by investigators outside of LASL. Donovan (14) computed the transient formation of a laminar vortex in a rectangular cavity where one wall was impulsively accelerated to a

constant velocity. Counter-rotating vortices appeared in the corners away from the moving wall, and steady state velocity components agreed with experimental data. Phillips (42) applied the method to impulsive and pressure-driven oscillating flows in a sudden expansion and in a tee. His interests were mainly numerical and although he recognized the existence of the truncation errors he did not try to remove them. Experimental comparisons were not available. Fernandez (17) used the MAC scheme to study pulsatile, incompressible flow in a bifurcation as a model of arterial flow. A dual grid system was mated at the junction of the bifurcation to carry the flow off at an angle. His solution showed a region of high shear stress at the inner (bifurcation) wall and a recirculation eddy at the outer wall which disappeared during part of the pulsation. These fluid phenomena were of interest in the formation of deposits in blood vessels.

The ICE (Implicit Continuous-Fluid Eulerian) method was presented in 1968 by Harlow and Amsden (27), then revised and improved in a later publication by the same authors (25). This is the only method found which could treat compressible flows at all speeds. The method combined concepts developed through the evolution of the compressible and incompressible methods discussed earlier. The differential forms of the equations were used and the pressure field was found implicitly as in MAC. The calculation of all other variables then could be performed explicitly. The Courant restriction applied to local fluid velocities. The method and program will be described in detail later. The authors performed in a truncation analysis for the one dimensional unsteady equations and suggested the addition of terms to correct these errors

and improve stability and accuracy. An example problem of an explosion leading to shock waves (compressible high speed flow) and a rising hot gas bubble (semi-compressible slow speed flow) was presented.

The problem of multispecie chemically reacting flows can be approached in two ways depending mainly on the kinetic models. One approach is to define a set of detailed kinetic reactions and to construct a model of blocks describing flow, reaction, and physical processes such as evaporation. The detail required in each block is specified according to current modeling capability and computer capacity. These blocks are then coupled together to describe the total process. The work of Edelman et al. (15) illustrates this approach. Another method is to write the full conservation equations in rigorous form with a global chemical reaction. Seider (47) has done this for dilute specie reactions. Both Edelman and Seider consider steady problems only. In his discussion of laminar flow reactions, Williams (56) points out that neither approach can be verified as the correct one. This thesis uses the second approach without the constraints of steady state or trace concentrations of reacting species.

This concludes the review of literature concepts and LASL developments pertinent to the present study. Several texts were used as sources of additional information. This includes Richtmeyer and Morton (44) and Von Rosenberg (52). The LASL monograph on fluid dynamics (26) and the excellent current text by Roache (45) have provided much physical and numerical insight. More information on the numerical methods developed at LASL may be found in the annotated bibliography by

Harlow (24). The text by Bird, Stewart, and Lightfoot (4) was used extensively to guide the mass transfer formulation. Much general information on boundary conditions and numerical methods was found in Schwab (46).

Chapter 3

FORMULATION OF THE PARTIAL DIFFERENTIAL EQUATIONS

A. Tensor Forms

The conservation equations which mathematically describe the class of unsteady two dimensional problems treated herein were listed in the introduction as equations (1-1) to (1-4). The stress tensor was equation (1-5). These equations will now be developed into the conventional notation of partial differential equations. A nondimensionalization will then be performed which is appropriate to the class of problems to be treated.

The equation for conservation of momentum (1-2) requires convariant differentiation of the viscous stress tensor. The differentiation is performed holding viscosity constant, although in the numerical problems the viscosity can be varied from point to point. This is equivalent to assuming that all spatial gradients of the viscosity coefficient make negligible contribution to the momentum viscous terms. Such an assumption is reasonable for many gas mixing problems. The differentiation gives

$$\tau^{ji}_{,i} = -\frac{1}{3}\mu g^{ji} v^{m}_{,mi} - \mu g^{mi} v^{j}_{,mi} \qquad (3\text{-}1)$$

Using the identity

$$g^{mi}v^{j}_{,mi} = g^{ji}v^{m}_{,mi} - \varepsilon^{jmp}\varepsilon_{pks}g^{ki}v^{s}_{,mi} \tag{3-2}$$

in equation (3-1) gives

$$\tau^{ji}_{,i} = -\frac{4}{3}\mu g^{ji}v^{m}_{,mi} + \mu\varepsilon^{jmp}\varepsilon_{pks}g^{ki}v^{s}_{,mi} \tag{3-3}$$

which is substituted into (1-2).

The equation for conservation of total energy also requires covariant differentiation of the product of the stress tensor and velocity. With a little manipulation this can be shown to give

$$\left(\tau^{mn}g_{nj}v^{j}\right)_{,m} = \frac{2}{3}\mu\left(v^{k}_{,k}v^{m}\right)_{,m} - \mu\left(v^{m}_{,q}v^{q}\right)_{,m} - \mu g^{mp}\left(v^{n}_{,p}v_{n}\right)_{,m} \tag{3-4}$$

The last term can be modified by the identity

$$v^{n}_{,p}v_{n} = \frac{1}{2}\left(v^{n}v_{n}\right)_{,p}$$

Hence

$$\left(\tau^{mn}g_{nj}v^{j}\right)_{,m} = \left[\frac{2}{3}\mu\left(v^{k}_{,k}v^{m}\right) - \mu\left(v^{m}_{,q}v^{q}\right) - \frac{1}{2}\mu g^{mp}\left(v^{n}g_{ns}v^{s}\right)_{,p}\right]_{,m} \tag{3-6}$$

This relationship is then substituted in (1-3).

The use of equations (3-3) and (3-6) allows the viscous stress to be removed from the momentum and energy equations, being replaced by velocities and velocity gradients. The conduction term may be written using Fourier's Law,

$$q^{j} = -kg^{jm}T_{,m} \tag{3-7}$$

For an ideal gas

$$I = C_{v}T \tag{3-8}$$

so

$$q^{j}_{,j} = - \frac{k}{C_V} g^{jm} I_{,jm} \tag{3-9}$$

This removes temperature as a variable in the energy equation.

The right hand side of the Kth species equation (1-4) may be expanded using

$$(n_K)^j = (j_K)^j + \rho_K v^j \tag{3-10}$$

For multicomponent mixtures of N ideal gases the mass diffusion term is

$$(j_K)^j = - \frac{C^2}{\rho} \sum_{R=1}^{N} M_R M_K D_{KR} g^{js} (X_R)_{,s} \tag{3-11}$$

The subscript R as used here does not denote a tensor quantity. The reaction rate is written as a generation term. For a second order reaction

$$(r_K) = k' \rho_K^A \rho_L^B \tag{3-12}$$

This is valid for the stoichiometry of A moles of specie K reacting with B moles of specie L. The rate coefficient k' in mass concentration units may be related to the rate in molar units by suitable ratios of molecular weights. Equations (3-10), (3-11), and (3-12) may be used in (1-4).

With all of the above substitutions, the final set of continuum equations in tensor form which mathematically describes unsteady compressible flow problems with multispecie diffusion and chemical reaction is

Mass:

$$\frac{\partial \rho}{\partial t} = - (\rho v^j)_{,j} \tag{3-13}$$

Momentum:

$$\frac{\partial(\rho v^j)}{\partial t} = - (\rho v^j v^i)_{,i} - g^{ji} P_{,i} + \frac{4}{3} \mu g^{ji} v^k_{,ki} - \mu \varepsilon^{jsp} \varepsilon_{pki} g^{km} v^i_{,ms} \tag{3-14}$$

Total Energy:

$$\frac{\partial(\rho E)}{\partial t} = - (\rho v^j E)_{,j} - (P v^j)_{,j} + \frac{k}{C_V} g^{jm} I_{,jm} + \left[\frac{2}{3} \mu \left(v^k_{,k} v^m \right) + \mu \left(v^m_{,q} v^q \right) + \frac{1}{2} \mu g^{mp} \left(v^n g_{ns} v^s_{,p} \right) \right]_{,m} \tag{3-15}$$

Multicomponent Species:

$$\frac{\partial(\rho_K)}{\partial t} = - \left(\rho_K v^j \right)_{,j} + \left[\frac{C}{M} \sum_{R=1}^{N} M_R M_K D_{KR} g^{js} (X_R)_{,s} \right]_{,j} + k' \rho_K^A \rho_L^B \tag{3-16a}$$

In some instances the multicomponent diffusion coefficient may be replaced by an effective binary diffusion coefficient as described in Bird et al. (4). The species K is assumed to move into a mixture which may be treated as a single fluid R containing no K. The mixture is denoted by R minus K (R-K). Then the conservation of specie mass is expressed by

Binary Species:

$$\frac{\partial(\rho_K)}{\partial t} = - \left(\rho_K v^j \right)_{,j} + \left[\frac{C}{M} M_K M_{R-K} \mathscr{D}_{K\ R-K} g^{js} (X_K)_{,s} \right]_{,j} + k' \rho_K^A \rho_L^B \tag{3-16b}$$

B. Dimensional Component Equations

The final set of dimensional component equations will be written in the right cylindrical system of coordinates to properly describe the tube flows studied herein. Hereafter sub- and superscripts are not used as tensor notation. Also hereafter overbars will denote dimensioned quantities, and r and z are also dimensioned.

$$\frac{\partial \bar{\rho}}{\partial \bar{t}} = -\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{\rho}\bar{v}_r\right) - \frac{\partial}{\partial z}\left(\bar{\rho}\bar{v}_z\right) \tag{3-17}$$

$$\frac{\partial(\bar{\rho}\bar{v}_r)}{\partial \bar{t}} = -\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{\rho}\bar{v}_r^2\right) - \frac{\partial}{\partial z}\left(\rho\bar{v}_r\bar{v}_z\right) - \frac{\partial \bar{P}}{\partial r}$$
$$+ \frac{4}{3}\bar{\mu}\frac{\partial}{\partial r}\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{v}_r\right) + \frac{\partial \bar{v}_z}{\partial z}\right] + \bar{\mu}\frac{\partial}{\partial z}\left[\frac{\partial \bar{v}_r}{\partial z} - \frac{\partial \bar{v}_z}{\partial r}\right] \tag{3-18}$$

$$\frac{\partial(\bar{\rho}\bar{v}_z)}{\partial \bar{t}} = -\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{\rho}\bar{v}_r\bar{v}_z\right) - \frac{\partial}{\partial z}\left(\bar{\rho}\bar{v}_z^2\right) - \frac{\partial \bar{P}}{\partial z}$$
$$+ \frac{4}{3}\bar{\mu}\frac{\partial}{\partial z}\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{v}_r\right) + \frac{\partial \bar{v}_z}{\partial r}\right] - \frac{\bar{\mu}}{r}\frac{\partial}{\partial r}\left[r\frac{\partial \bar{v}_r}{\partial z} - r\frac{\partial \bar{v}_z}{\partial r}\right] \tag{3-19}$$

$$\frac{\partial(\bar{\rho}\bar{E})}{\partial \bar{t}} = -\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{\rho}\bar{v}_r\bar{E}\right) - \frac{\partial}{\partial z}\left(\bar{\rho}\bar{v}_z\bar{E}\right) + \frac{1}{r}\frac{\partial}{\partial r}\left\{r\left[-\bar{P}\bar{v}_r + \frac{\bar{k}}{\bar{C}_v}\frac{\partial \bar{I}}{\partial r}\right.\right.$$
$$\left. - \frac{2}{3}\bar{\mu}\bar{v}_r\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{v}_r\right) + \frac{\partial \bar{v}_z}{\partial z}\right] + \frac{1}{2}\bar{\mu}\frac{\partial}{\partial r}\left(2\bar{v}_r^2 + \bar{v}_z^2\right) + \bar{\mu}\bar{v}_z\frac{\partial \bar{v}_r}{\partial z}\right]\right\}$$
$$+ \frac{\partial}{\partial z}\left\{-\bar{P}\bar{v}_z + \frac{\bar{k}}{\bar{C}_v}\frac{\partial \bar{I}}{\partial z} - \frac{2}{3}\bar{\mu}\bar{v}_z\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\bar{v}_r\right) + \frac{\partial \bar{v}_z}{\partial z}\right]\right.$$
$$\left. + \frac{1}{2}\bar{\mu}\frac{\partial}{\partial z}\left(\bar{v}_r^2 + 2\bar{v}_z^2\right) + \bar{\mu}\bar{v}_r\frac{\partial \bar{v}_z}{\partial r}\right\} \tag{3-20}$$

$$\frac{\partial(\overline{\rho}_K)}{\partial \overline{t}} = -\frac{1}{r}\frac{\partial}{\partial r}(r\overline{\rho}_K\overline{v}_r) - \frac{\partial}{\partial z}(\overline{\rho}_K\overline{v}_z) + \frac{1}{r}\frac{\partial}{\partial r}\left[r\frac{\overline{C}}{\overline{M}}\sum_{R=1}^{N}\overline{M}_R\overline{M}_K\overline{D}_{KR}\frac{\partial X_R}{\partial r}\right]$$

$$+\frac{\partial}{\partial z}\left[\frac{\overline{C}}{\overline{M}}\sum_{R=1}^{N}\overline{M}_R\overline{M}_K\overline{D}_{KR}\frac{\partial X_R}{\partial z}\right] + \overline{k}'\overline{\rho}_K^{A-B}\overline{\rho}_L \qquad (3\text{-}21a)$$

$$\frac{\partial(\overline{\rho}_K)}{\partial \overline{t}} = -\frac{1}{r}\frac{\partial}{\partial r}(r\overline{\rho}_K\overline{v}_r) - \frac{\partial}{\partial z}(\overline{\rho}_K\overline{v}_z) + \frac{1}{r}\frac{\partial}{\partial r}\left[r\frac{\overline{C}}{\overline{M}}\overline{M}_{R-K}\overline{M}_K\overline{\mathscr{D}}_{K\ R-K}\frac{\partial X_K}{\partial r}\right]$$

$$+\frac{\partial}{\partial z}\left[\frac{\overline{C}}{\overline{M}}\overline{M}_{R-K}\overline{M}_K\overline{\mathscr{D}}_{K\ R-K}\frac{\partial X_K}{\partial z}\right] + \overline{k}'\overline{\rho}_K^{A-B}\overline{\rho}_L \qquad (3\text{-}21b)$$

Two additional equations are needed. The total energy is composed of internal and kinetic energy. Then the internal energy is calculated by

$$\overline{I} = \overline{E} - \frac{1}{2}\left(\overline{v}_r^2 + \overline{v}_z^2\right) \qquad (3\text{-}22)$$

An equation of state for an ideal gas is

$$\overline{P} = (\gamma - 1)\overline{\rho}\overline{I} \qquad (3\text{-}23)$$

The set from equation (3-17) to (3-23) includes all equations necessary to describe continuum flow. Boundary conditions will be treated later.

C. Dimensionless Component Equations

The partial differential equations will now be made dimensionless. A set of characteristic values are selected as follows:

Length	$= \overline{R}_W$	the tube outer radius
Velocity	$= \overline{V}$	usually a known input velocity
Time	$= \overline{t}_{\sim}$	the cycle time or period for oscillating flows. If the flows are not oscillatory, this may be set to one, or any other time characteristic of the problem

State Conditions = $\overline{\rho}^*$, $\overline{P}^*$, $\overline{M}^*$, $\overline{I}^*$, $\overline{T}^*$, $\overline{C}^*$ the conditions of a reference stream.

These characteristic quantities are used to define dimensionless variables.

$$t = \overline{t}/\overline{t}_{\sim} \tag{3-24}$$

$$V = \overline{v}_r/\overline{V} \tag{3-25}$$

$$U = \overline{v}_z/\overline{V} \tag{3-26}$$

$$R = r/\overline{R}_W \tag{3-27}$$

$$Z = z/\overline{R}_W \tag{3-28}$$

$$\rho = \overline{\rho}/\overline{\rho}^* \tag{3-29}$$

$$\rho_K = \overline{\rho}_K/\overline{\rho}^* \tag{3-30}$$

$$E = \overline{E}/\overline{V}^2 \tag{3-31}$$

$$I = \overline{I}/\overline{V}^2 \tag{3-32}$$

$$k' = \overline{k}'\overline{t}_{\sim} \Big/ (p^*)^{1-A-B} \tag{3-33}$$

$$P = (\overline{P} - \overline{P}^*)/\overline{P}^* \tag{3-34}$$

$$T = (\overline{T} - \overline{T}^*)/\overline{T}^* \tag{3-35}$$

$$C = \overline{C}/\overline{C}^* \tag{3-36}$$

$$M = \overline{M}/\overline{M}^* \tag{3-37}$$

The temperature and internal energy are related by

$$\overline{I}^* = \overline{R}_C\overline{T}^*/[(\gamma - 1)\overline{M}^*] \tag{3-38}$$

This permits definition of a reference internal energy if a reference thermodynamic state is specified.

When the set (3-24) through (3-37) is substituted into the component equations, several dimensionless groups arise. These are defined as follows:

Euler Number $$N_{Eu} = \left[\frac{\bar{P}^*}{\bar{\rho}^* \bar{V}^2}\right] \qquad (3\text{-}39)$$

Prandtl Number $$N_{Pr} = \left[\frac{\bar{C}_p \bar{\mu}}{\bar{k}}\right] \qquad (3\text{-}40)$$

Reynolds Number $$N_{Re} = \left[\frac{\bar{\rho}^* \bar{R}_W \bar{V}}{\bar{\mu}}\right] \qquad (3\text{-}41)$$

Schmidt Number $$N_{Sc} = \left[\frac{\bar{\mu}}{\bar{\rho}^* \bar{D}_{KR}}\right] \qquad (3\text{-}42a)$$

or $$= \left[\frac{\bar{\mu}}{\bar{\rho}^* \bar{\mathcal{D}}_{K\ R\text{-}K}}\right] \qquad (3\text{-}42b)$$

Modified Strouhal Number $$N_{S\ell} = \left[\frac{\bar{R}_W}{\bar{V}\bar{t}_\sim}\right] \qquad (3\text{-}43)$$

The Strouhal Number as normally defined is

$$N_{S\ell} = 2\pi \bar{f} \bar{R}_W / \bar{V}$$

It appears in cyclic phenomena such as acoustics, and it ratios the action time of a characteristic velocity acting over a characteristic distance to the time required for a cyclic perturbation. Since $\bar{t}_\sim$ corresponds to $1/\bar{f}$, then $N_{S\ell}$ is $1/2\pi$ times the normal definition of a Strouhal Number.

With these definitions, the dimensionless component equations may be written,

Mass:

$$N_{S\ell}\frac{\partial\rho}{\partial t} = -\frac{1}{R}\frac{\partial}{\partial R}(R\rho V) - \frac{\partial}{\partial Z}(\rho U) \tag{3-44}$$

R-Momentum:

$$N_{S\ell}\frac{\partial(\rho V)}{\partial t} = -\frac{1}{R}\frac{\partial}{\partial R}(R\rho V^2) - \frac{\partial}{\partial Z}(\rho VU) - N_{Eu}\frac{\partial P}{\partial R} + \frac{1}{N_{Re}}\left\{\frac{4}{3}\frac{\partial}{\partial R}\left[\frac{1}{R}\frac{\partial}{\partial R}(RV) + \frac{\partial U}{\partial Z}\right] + \frac{\partial}{\partial Z}\left[\frac{\partial V}{\partial Z} - \frac{\partial U}{\partial R}\right]\right\} \tag{3-45}$$

Z-Momentum:

$$N_{S\ell}\frac{\partial(\rho U)}{\partial t} = -\frac{1}{R}\frac{\partial}{\partial R}(R\rho VU) - \frac{\partial}{\partial Z}(\rho U^2) - N_{Eu}\frac{\partial P}{\partial Z} + \frac{1}{N_{Re}}\left\{\frac{4}{3}\frac{\partial}{\partial Z}\left[\frac{1}{R}\frac{\partial}{\partial R}(RV) + \frac{\partial U}{\partial Z}\right] - \frac{1}{R}\frac{\partial}{\partial R}R\left[\left(\frac{\partial V}{\partial Z} - \frac{\partial U}{\partial R}\right)\right]\right\} \tag{3-46}$$

Total Energy:

$$\begin{aligned} N_{S\ell}\frac{\partial(\rho E)}{\partial t} = {} & -\frac{1}{R}\frac{\partial}{\partial R}(R\rho VE) - \frac{\partial}{\partial Z}(\rho UE) \\ & + \frac{1}{R}\frac{\partial}{\partial R}\left\{R\left[-N_{Eu}(P+1)V + \frac{1}{N_{Re}}\left[\frac{\gamma}{N_{Pr}}\frac{\partial I}{\partial R}\right.\right.\right. \\ & \left.\left.\left. - \frac{2}{3}\left(\frac{1}{R}\frac{\partial}{\partial R}(RV) + \frac{\partial U}{\partial Z}\right)V + \frac{1}{2}\frac{\partial}{\partial R}(2V^2 + U^2) + U\frac{\partial V}{\partial Z}\right]\right]\right\} \\ & + \frac{\partial}{\partial Z}\left\{-N_{Eu}(P+1)U + \frac{1}{N_{Re}}\left[\frac{\gamma}{N_{Pr}}\frac{\partial I}{\partial Z}\right.\right. \\ & \left.\left. - \frac{2}{3}\left(\frac{1}{R}\frac{\partial}{\partial R}(RV) + \frac{\partial U}{\partial Z}\right)U + \frac{1}{2}\frac{\partial}{\partial Z}(V^2 + 2U^2) + V\frac{\partial U}{\partial R}\right]\right\} \end{aligned} \tag{3-47}$$

Multicomponent Species:

$$N_{S\ell}\frac{\partial(\rho_K)}{\partial t} = -\frac{1}{R}\frac{\partial}{\partial R}(R\rho_K V) - \frac{\partial}{\partial Z}(\rho_K U) + \frac{1}{R}\frac{\partial}{\partial R}\left[R\frac{C}{M}\sum_{R=1}^{N}\frac{M_R M_K}{N_{Re}N_{Sc}}\frac{\partial X_R}{\partial R}\right] + \frac{\partial}{\partial Z}\left[\frac{C}{M}\sum_{R=1}^{N}\frac{M_R M_K}{N_{Re}N_{Sc}}\frac{\partial X_R}{\partial Z}\right] + N_{S\ell}k'\rho_K^A\rho_L^B \tag{3-48}$$

Binary Species:

$$N_{S\ell}\frac{\partial(\rho_K)}{\partial t} = -\frac{1}{R}\frac{\partial}{\partial R}(R\rho_K V) - \frac{\partial}{\partial Z}(\rho_K U) + \frac{1}{R}\frac{\partial}{\partial R}\left[R\,\frac{C}{M}\,\frac{M_{R-K}M_K}{N_{Re}N_{Sc}}\,\frac{\partial X_K}{\partial R}\right]$$

$$+\frac{\partial}{\partial Z}\left[\frac{C}{M}\,\frac{M_{R-K}M_K}{N_{Re}N_{Sc}}\,\frac{\partial X_K}{\partial Z}\right] + N_{S\ell}k'\rho_K^A\rho_L^B \qquad (3\text{-}49)$$

Internal Energy:

$$I = E - \frac{1}{2}[V^2 + U^2] \qquad (3\text{-}50)$$

Equation of State:

$$P = \frac{(\gamma - 1)}{N_{Eu}}\rho I - 1 \qquad (3\text{-}51)$$

A few points should be made about these equations. The equations as shown are in conservative form, relating the change of a quantity at a point to fluxes and forces acting on an infinitesmal surface enclosing the point. Cheng (11) has stated that only this form gives satisfactory accuracy in numerical solutions of the full Navier-Stokes equations. A complete energy equation using temperature as the dependent variable can not be written conservatively. Hence total and internal energy is used.

Isothermal incompressible flow does not require an energy equation. But calculations of compressible flows must include an energy equation, and the energy couples with the fluid dynamics through the equation of state. The state equation used herein applies to ideal gases, and it and the energy equation are bypassed for incompressible calculations. Only equation (3-51) need be changed to accommodate non-ideal fluids. Any equation of state which properly describes the fluid may be used. The LASL monograph (26) discusses some other state

equations, including one describing the behavior of a normally solid material flowing under ultra-high velocity impact.

The next chapter depicts the finite differencing of the equations and outlines the method of their solution.

Chapter 4

FORMULATION OF THE FINITE DIFFERENCE EQUATIONS

A. The ICE Cell and Computational Mesh

The continuum equations are converted to discrete equations by casting a mesh over the field of interest and defining the variables at select locations on the mesh. For the ICE system, the mesh consists of a series of cells, dimensioned ΔR and ΔZ in the radial and axial directions, respectively. These cells are actually cross-sections of annular rings where angular variations are presumed negligible. A typical cell is shown in figure 1. The cell coordinates are indexed at its center, i counting cell rows increasing in the radial direction toward the wall, and j counting cell columns increasing in the downstream axial direction. The axial and radial velocities are defined on the middle of the cell walls, and all the other variables are defined at the cell center. This system is also used in the MAC scheme. The cell aspect ratio is

$$R_A = \Delta R/\Delta Z \qquad (4\text{-}1)$$

There is no constraint that the cells be square.

The temporal and spatial coordinates of each variable is denoted by a superscript and two subscripts. The superscript denotes time, with n being the current time t, and $n + 1$ being advanced time

$t + \Delta t$. Current variables are called explicit, and advanced time variables are called implicit. The two subscripts denote radial and axial position, respectively. A typical cell-centered variable such as density in the (i, j)th cell is $\rho^n_{i,j}$. The downstream axial velocity on that cell is $U^n_{i,j+1/2}$, and the radial velocity closer to the centerline on that cell is $V^n_{i-1/2,j}$. If a value for a variable is required at a location where it is not defined, a simple average is used. Thus

$$U^n_{i,j} = \frac{1}{2}\left(U^n_{i,j-1/2} + U^n_{i,j+1/2}\right) \tag{4-2}$$

$$\rho^n_{i+1/2,j-1/2} = \frac{1}{4}\left(\rho^n_{i+1,j} + \rho^n_{i,j} + \rho^n_{i+1,j-1} + \rho^n_{i,j-1}\right) \tag{4-3}$$

When a differential is required at a location such that averaging is necessary, the rule is: form the differential, then perform the averaging. For example,

$$\left.\frac{\partial \rho}{\partial Z}\right|^n_{i+1/2,j+1/2} \approx \frac{\rho^n_{i+1/2,j} - \rho^n_{i+1/2,j-1}}{\Delta Z} = \frac{\rho^n_{i+1,j} + \rho^n_{i,j} - \rho^n_{i+1,j-1} - \rho^n_{i,j-1}}{2\Delta Z} \tag{4-4}$$

The placement of velocities at locations different than the other variables has a number of advantages. First, it provides a convenient arrangement for defining centered velocity differences. Consider the spatial first derivative, $\partial U/\partial Z$, to be evaluated at the point (i, j). Presume that all variables are defined at (i, j) including U. Then the differential may be approximated at least three ways by Taylor Series expansions around the point (i, j).

$$\text{Forward} \quad \left.\frac{\partial U}{\partial Z}\right|^{n}_{i,j} \approx \frac{U^{n}_{i,j+1} - U^{n}_{i,j}}{\Delta Z} + \text{terms } O(\Delta Z) \tag{4-5}$$

$$\text{Backward} \quad \left.\frac{\partial U}{\partial Z}\right|^{n}_{i,j} \approx \frac{U^{n}_{i,j} - U^{n}_{i,j-1}}{\Delta Z} + \text{terms } O(\Delta Z) \tag{4-6}$$

$$\text{Centered} \quad \left.\frac{\partial U}{\partial Z}\right|^{n}_{i,j} \approx \frac{U^{n}_{i,j+1} - U^{n}_{i,j-1}}{2\Delta Z} + \text{terms } O(\Delta Z)^{2} \tag{4-7}$$

The centered difference has higher accuracy, which is desirable. But the two values of U are separated by $2\Delta Z$, and the value of U at the point (i, j) is not used at all. By defining variables on the cell as shown in figure 1 the cell centered differential becomes

$$\left.\frac{\partial U}{\partial Z}\right|^{n}_{i,j} \approx \frac{U^{n}_{i,j+1/2} - U^{n}_{i,j-1/2}}{\Delta Z} + \text{terms } O(\Delta Z)^{2} \tag{4-8}$$

and the values of U are separated by only ΔZ. Since the values are closer to the point where the differential is desired, this approximation is better than (4-7). First derivatives of velocity are important quantities. With all other variables defined on the cell centers, the opportunity to use the form (4-8) occurs frequently.

Another advantage of the ICE variable placement is improved stability. Richtmeyer and Morton (44) show that skew velocity placement such as in the ICE method provides less restriction on stable values of Δt and ΔZ than does defining both velocities at the same point. The skew scheme is often used for hyperbolic equations.

Finally, the variable placement on the ICE cell provides an aid to visualizing the physical aspects of the problems. The cell centered

properties describe the "contents" of the cell. Since the velocities are defined on the walls, they can be envisioned as carrying the cell contents through the cell walls.

The cells fit together to form a grid on the field of interest to the calculation. Such cells are called interior cells. An additional row or column of cells is added to each boundary. These cells are artificial, in that they exist only to apply boundary conditions to the interior cells. Figure 2 shows the calculation mesh with the boundary cells. Four types of boundaries are shown: wall, symmetrical centerline, input, and output. These will be discussed later. Since the first row of cells $i = 1$ is a boundary row, and the tube centerline is at $R = 0$, the radial distances to the inside, center, and outside of cell (i, j) are

$$R_{i-1/2} = (i - 2)\Delta R \tag{4-9}$$

$$R_i = \frac{1}{2}(2i - 3)\Delta R \tag{4-10}$$

$$R_{i+1/2} = (i - 1)\Delta R \tag{4-11}$$

B. Development of the Poisson Equation for Pressure

With the cell and grid system defined, the finite difference approximation of the mass equation is written at the point (i, j) as

$$N_{S\ell}\frac{\left(\rho_{i,j}^{n+1}-\rho_{i,j}^{n}\right)}{\Delta t}=-\theta\frac{\left(R_{i+1/2}\rho_{i+1/2,j}^{n+1}V_{i+1/2,j}^{n+1}-R_{i-1/2}\rho_{i-1/2,j}^{n+1}V_{i-1/2,j}^{n+1}\right)}{R_i\Delta R}$$

$$-\theta\frac{\left(\rho_{i,j+1/2}^{n+1}U_{i,j+1/2}^{n+1}-\rho_{i,j-1/2}^{n+1}U_{i,j-1/2}^{n+1}\right)}{\Delta Z}$$

$$-(1-\theta)\frac{\left(R_{i+1/2}\rho_{i+1/2,j}^{n}V_{i+1/2,j}^{n}-R_{i-1/2}\rho_{i-1/2,j}^{n}V_{i-1/2,j}^{n}\right)}{R_i\Delta R}$$

$$-(1-\theta)\frac{\left(\rho_{i,j+1/2}^{n}U_{i,j+1/2}^{n}-\rho_{i,j-1/2}^{n}U_{i,j-1/2}^{n}\right)}{\Delta Z}+(\beta_M)_{i,j}^{n} \qquad (4\text{-}12)$$

Note that the mass fluxes are repeated, first implicitly with superscript n+1, then explicitly with superscript n. The implicit and explicit groupings have coefficients θ and $(1-\theta)$, respectively. Values of θ range from 0 to 1 with the value held constant over time and position for a particular numerical computation. θ acts to proportion the amount of implicit versus explicit mass fluxing used in computing the change of cell centered density with time. If $\theta = .5$, the equation is time-centered and certain truncation errors vanish. The term β_M is a correction which appears as an added diffusion term. It may be used to improve the scheme's stability and accuracy by removing certain truncation errors, or if stability is a problem the β_M term can provide additional stability at the expense of accuracy. β corrections will be discussed in the next chapter.

Before finite differencing the component momentum equations, it is necessary to define a new variable, which Harlow and Amsden (27) have called the hybrid function $\hat{P}$. The hybrid function is formed from the equation of state, which has the general functional form

$$P = P(\rho, I) \tag{4-13}$$

This function may be expanded over intervals around the state (ρ, I) to give

$$P(\rho + \Delta\rho,\ I + \Delta I) = P(\rho, I) + \left(\frac{\partial P}{\partial \rho}\right)_I \Delta\rho + \left(\frac{\partial P}{\partial I}\right)_P \Delta I + \text{terms } O(\Delta^2) \tag{4-14}$$

The equation of state in finite difference form is

$$P^n_{i,j} = \frac{(\gamma - 1)^n_{i,j}}{N_{Eu}} \rho^n_{i,j} I^n_{i,j} - 1 \tag{4-15}$$

Equation (4-14) may be written at the point (i,j), and if the Taylor expansion is understood to be over the time domain, the differences $\Delta\rho$ and ΔI are implicit minus explicit values. Defining the left side of equation (4-14) as the hybrid function $\tilde{P}$, a tentative implicit pressure, and using (4-15) to evaluate the partial derivatives,

$$\tilde{P}_{i,j} = P^n_{i,j} + A^n_{i,j}\left(\rho^{n+1}_{i,j} - \rho^n_{i,j}\right) + \left(\frac{(\gamma - 1)^n_{i,j}}{N_{Eu}}\right) \rho^n_{i,j}\left(I^{n+1}_{i,j} - I^n_{i,j}\right) \tag{4-16}$$

The series is truncated after the first order terms, and the isothermal speed of sound squared is

$$A^n_{i,j} = \left(\frac{\partial P}{\partial \rho}\right)^2_I \Bigg|^n_{i,j} = \left(\frac{(\gamma - 1)^n_{i,j}}{N_{Eu}}\right) I^n_{i,j} \tag{4-17}$$

The LASL ICE method excludes the first order difference $\left(I^{n+1}_{i,j} - I^n_{i,j}\right)$. Although most calculations performed herein also discard this difference, some numerical experimentation is done with the difference included. It will therefore be retained in the finite differencing. Equation (4-16) may be solved for the implicit density. This gives

$$\rho_{i,j}^{n+1} = \left(\frac{\tilde{P}_{i,j} - P_{i,j}}{A_{i,j}^{n}}\right) + \rho_{i,j}^{n}\left(2 - \frac{I_{i,j}^{n+1}}{I_{i,j}^{n}}\right) \qquad (4\text{-}18)$$

With the hybrid function $\tilde{P}$ defined, the momentum equations may be finite differenced. For both the R and Z component equations all the flux and shear terms are written explicitly. Only the pressure term is written both implicitly and explicitly, and the hybrid function is used as the implicit pressure. A proportioning constant φ appears in the momentum equations in the same manner as θ in the mass equation.

The radial momentum equation is written at the point (i-1/2,j) as

$$N_{S\ell}\left[\frac{\rho_{i-1/2,j}^{n+1} V_{i-1/2,j}^{n+1} - \rho_{i-1/2,j}^{n} V_{i-1/2,j}^{n}}{\Delta t}\right] = - N_{Eu}\varphi\left[\frac{\tilde{P}_{i,j} - \tilde{P}_{i-1,j}}{\Delta R}\right] - N_{Eu}(1-\varphi)\left[\frac{P_{i,j}^{n} - P_{i-1,j}^{n}}{\Delta R}\right] + B'_{i-1/2,j} \qquad (4\text{-}19)$$

$B'_{i-1/2,j}$ is a collection of explicit terms for momentum fluxes, shear terms, and a truncation error correction β_{VR}. Likewise, the axial momentum equation is written at the point (i,j-1/2) as

$$N_{S\ell}\left[\frac{\rho_{i,j-1/2}^{n+1} U_{i,j-1/2}^{n+1} - P_{i,j-1/2}^{n} U_{i,j-1/2}^{n}}{\Delta t}\right] = - N_{Eu}\varphi\,\frac{\tilde{P}_{i,j} - \tilde{P}_{i,j-1}}{\Delta Z} - N_{Eu}(1-\varphi)\left[\frac{P_{i,j}^{n} - P_{i,j-1}^{n}}{\Delta Z}\right] + D'_{i,j-1/2} \qquad (4\text{-}20)$$

$D'_{i,j-1/2}$ collects explicit flux and shear terms plus a correction β_{VZ} for the axial momentum equation.

A Poisson equation for $\tilde{P}$ is formed as follows from equations (4-12), (4-18), (4-19), and (4-20). The coefficients $(1 - \theta)$ in the mass equation (4-12) are split into two coefficients, 1 and $(-\theta)$. Then the explicit terms multiplied by $(-\theta)$ are grouped with the implicit terms multiplied by θ to produce differences such as

$$\left(R_{i+1/2}\rho^{n+1}_{i+1/2,j}V^{n+1}_{i+1/2,j} - R_{i+1/2}P^{n}_{i+1/2,j}V^{n}_{i+1/2,j}\right)$$

That is, these terms are differences in time at the same spatial position. An identical difference is formed at coordinates (i-1/2,j). Differences of ρU at coordinates (i,j+1/2) and (i,j-1/2) are also formed. It can be seen that if equation (4-19) is written at the point (i+1/2,j) and multiplied by $R_{i+1/2}$, the left hand side of the result contains the difference shown above times $N_{S\ell}/\Delta t$. This is repeated at the point (i-1/2,j). Equation (4-20) is used at points (i,j+1/2) and (i,j-1/2). Thus with a little manipulation, the momentum equations may be substituted into the mass equation to remove advanced time mass fluxes while introducing the hybrid function $\tilde{P}$. After these substitutions are made, the only remaining implicit density is in the time difference of the mass equation. This may be removed using (4-18). The result is

$$\tilde{P}_{i,j} = \frac{\left[\frac{\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2}\right]\left[\frac{R_{i+1/2}\tilde{P}_{i+1,j} + R_{i-1/2}\tilde{P}_{i-1,j}}{R_i(\Delta R)^2} + \frac{\tilde{P}_{i,j+1} + \tilde{P}_{i,j-1}}{(\Delta Z)^2}\right] + G^n_{i,j} - \rho^n_{i,j}\left[1 - \frac{I^{n+1}_{i,j}}{I^n_{i,j}}\right]}{\left[\frac{1}{A^n_{i,j}} + \frac{2\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2}\right]\left(\frac{1}{(\Delta R)^2} + \frac{1}{(\Delta Z)^2}\right)} \qquad (4\text{-}21)$$

The term $G^n_{i,j}$ groups a number of explicit terms and incorporates the corrector β_M.

C. Final Finite Difference Forms

The finite difference equations are shown in their final form. The radii are written using equations (4-9) to (4-11), and ΔZ is removed using the cell aspect ratio, equation (4-1). When a variable is required at a location where it is not defined, the necessary averages are formed. Most averages are bracketed and easy to detect. Coefficients are extracted wherever possible.

Variables which collect explicit terms are written below. The velocity divergence is used several places and is denoted by $Q^n_{i,j}$.

$$Q^n_{i,j} = \frac{8}{3}\left[\left(\frac{i-1}{2i-3}\right)V^n_{i+1/2,j} - \left(\frac{i-2}{2i-3}\right)V^n_{i-1/2,j} + \frac{1}{2}R_A\left(U^n_{i,j+1/2} - U^n_{i,j-1/2}\right)\right] \qquad (4\text{-}22)$$

The two variables used in the momentum equations are

$$B^n_{i-1/2,j} = (\Delta R)B'_{i-1/2,j} = \frac{1}{2} V^n_{i-1/2,j} \left[\left(\frac{2i-5}{i-2} \right) \rho^n_{i-1,j} V^n_{i-3/2,j} - \left(\frac{2i-3}{i-2} \right) \rho^n_{i,j} V^n_{i+1/2,j} \right]$$

$$+ \frac{R_A}{16} \left[\left(\rho^n_{i,j} + \rho^n_{i-1,j} + \rho^n_{i,j-1} + \rho^n_{i-1,j-1} \right) \left(V^n_{i-1/2,j} + V^n_{i-1/2,j-1} \right) \left(U^n_{i,j-1/2} + U^n_{i-1,j-1/2} \right) \right.$$

$$\left. - \left(\rho^n_{i,j+1} + \rho^n_{i-1,j+1} + \rho^n_{i,j} + \rho^n_{i-1,j} \right) \left(V^n_{i-1/2,j+1} + V^n_{i-1/2,j} \right) \left(U^n_{i,j+1/2} + U^n_{i-1,j+1/2} \right) \right]$$

$$+ \left[\frac{2}{\Delta R} \right] \left[\frac{1}{(N_{Re})^n_{i,j} + (N_{Re})^n_{i-1,j}} \right] \left[\left(Q^n_{i,j} - Q^n_{i-1,j} \right) + R_A^2 \left(V^n_{i-1/2,j+1} + V^n_{i-1/2,j-1} - 2V^n_{i-1/2,j} \right) \right.$$

$$\left. - R_A \left(U^n_{i,j+1/2} - U^n_{i-1,j+1/2} - U^n_{i,j-1/2} + U^n_{i-1,j-1/2} \right) \right] + \Delta R (\beta_{VR})^n_{i-1/2,j} \qquad (4\text{-}23)$$

and

$$D^n_{i,j-1/2} = \Delta R \, D'_{i,j-1/2} = \frac{1}{8} \left[\left(\frac{i-2}{2i-3} \right) \left(\rho^n_{i,j} + \rho^n_{i-1,j} + \rho^n_{i,j-1} + \rho^n_{i-1,j-1} \right) \left(V^n_{i-1/2,j} + V^n_{i-1/2,j-1} \right) \right.$$

$$\left(U^n_{i,j-1/2} + U^n_{i-1,j-1/2} \right) - \left(\frac{i-1}{2i-3} \right) \left(\rho^n_{i+1,j} + \rho^n_{i,j} + \rho^n_{i+1,j-1} + \rho^n_{i,j-1} \right)$$

$$\left. \left(V^n_{i+1/2,j} + V^n_{i+1/2,j-1} \right) \left(U^n_{i+1,j-1/2} + U^n_{i,j-1/2} \right) \right]$$

$$+ R_A U^n_{i,j-1/2} \left(\rho^n_{i,j-1} U^n_{i,j-3/2} - \rho^n_{i,j} U^n_{i,j+1/2} \right) + \left[\frac{2}{\Delta R} \right] \left[\frac{1}{(N_{Re})^n_{i,j} + (N_{Re})^n_{i,j-1}} \right] \left\{ R_A \left(Q^n_{i,j} - Q^n_{i,j-1} \right) \right.$$

(this eq. is continued from previous page)

$$+ 2R_A \left[\left(\frac{i-2}{2i-3} \right) \left(V^n_{i-1/2,j} - V^n_{i-1/2,j-1} \right) - \left(\frac{i-1}{2i-3} \right) \left(V^n_{i+1/2,j} - V^n_{i+1/2,j-1} \right) \right]$$

$$+ 2 \left[\left(\frac{i-1}{2i-3} \right) U^n_{i+1,j-1/2} + \left(\frac{i-2}{2i-3} \right) U^n_{i-1,j-1/2} - U^n_{i,j-1/2} \right] \Bigg\} + \Delta R (\beta_{VZ})^n_{i,j-1/2} \qquad (4\text{-}24)$$

The term which appears in the $\tilde{P}$ Poisson equation is

$$G^n_{i,j} = \frac{P^n_{i,j}}{A^n_{i,j}} + \left[\frac{2\theta(\Delta t)^2}{N^2_{S\ell}(\Delta R)^2} \right] \Bigg\{ N_{Eu}(1-\varphi) \Bigg[\left(\frac{i-1}{2i-3} \right) P^n_{i+1,j} + \left(\frac{i-2}{2i-3} \right) P^n_{i-1,j} - P^n_{i,j}$$

$$+ \frac{1}{2} R_A^2 \left(P^n_{i,j+1} + P^n_{i,j-1} - 2P^n_{i,j} \right) \Bigg] + \left(\frac{i-2}{2i-3} \right) B^n_{i-1/2,j} - \left(\frac{i-1}{2i-3} \right) B^n_{i+1/2,j}$$

$$+ \frac{1}{2} R_A \left(D^n_{i,j-1/2} - D^n_{i,j+1/2} \right) \Bigg\} + \left[\frac{\Delta t}{N_{S\ell} \Delta R} \right] \Bigg\{ \left(\frac{i-2}{2i-3} \right) \left(\rho^n_{i,j} + \rho^n_{i-1,j} \right) V^n_{i-1/2,j}$$

$$- \left(\frac{i-1}{2i-3} \right) \left(\rho^n_{i+1,j} + \rho^n_{i,j} \right) V^n_{i+1/2,j} + \frac{1}{2} R_A \Big[\left(\rho^n_{i,j} + \rho^n_{i,j-1} \right) U^n_{i,j-1/2}$$

$$- \left(\rho^n_{i,j+1} + \rho^n_{i,j} \right) U^n_{i,j+1/2} \Big] \Bigg\} + \frac{\Delta t}{N_{S\ell}} (\beta_M)^n_{i,j} \qquad (4\text{-}25)$$

The explicit momentum fluxes in equations (4-23) and (4-24) are written using ZIP differencing, which is a conservative method discussed in the LASL papers. It defines the finite difference momentum flux through a cell wall so that the flux is arithmetically the same whether

viewed from the donor or the recipient cell. ZIP differencing has fewer truncation errors to be corrected by β_{VR} and β_{VZ}.

The Poisson equation for $\tilde{P}$ was solved by point successive over relaxation (SOR) and also alternating direction implicit methods. These are discussed in detail in Appendix A. The final form of the $\tilde{P}$ equation is

$$\tilde{P}_{i,j} = \frac{\left[\frac{2\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2(\Delta R)^2}\right]\left[\left(\frac{i-1}{2i-3}\right)\tilde{P}_{i+1,j} + \left(\frac{i-2}{2i-3}\right)\tilde{P}_{i-1,j} + \frac{1}{2}R_A^2\left(\tilde{P}_{i,j+1} + \tilde{P}_{i,j-1}\right)\right] + G^n_{i,j} - \rho^n_{i,j}\left[1 - \frac{I^{n+1}_{i,j}}{I^n_{i,j}}\right]}{\left[\frac{1}{A^n_{i,j}} + \left[\frac{2\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2(\Delta R)^2}\right]\left(1 + R_A^2\right)\right]} \tag{4-26}$$

The functional SOR form with relaxation coefficient α is

$$\tilde{P}^{Q+1}_{i,j} = (1-\alpha)\,\tilde{P}^Q_{i,j} + \alpha\mathcal{L}\left(\tilde{P}^{Q+1}_{i-1,j}, \tilde{P}^{Q+1}_{i,j-1}, \tilde{P}^Q_{i+1,j}, \tilde{P}^Q_{i,j+1}, G^n_{i,j}, \rho^n_{i,j}, I^{n+1}_{i,j}, I^n_{i,j}\right) \tag{4-27}$$

For ADI the equation is made into a pseudo time-dependent equation with variable χ as the pseudo time. Two equations are required to bridge an increment $\Delta\chi$.

$$\left[\frac{2\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2(\Delta R)^2}\left(\frac{i-2}{2i-3}\right)\right]\tilde{P}^{Q+1/2}_{i-1,j}+\left[-\frac{2}{\Delta X}-\frac{2\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2(\Delta R)^2}\right]\tilde{P}^{Q+1/2}_{i,j}+\left[\frac{2\theta\varphi(\Delta t)^2 N_{Eu}}{N_{S\ell}^2(\Delta R)^2}\left(\frac{i-1}{2i-3}\right)\right]\tilde{P}^{Q+1/2}_{i+1,j}=$$

$$-\left[\frac{\theta\varphi(\Delta t)^2 N_{Eu}R_A^2}{N_{S\ell}^2(\Delta R)^2}\right]\left(\tilde{P}^{Q}_{i,j+1}+\tilde{P}^{Q}_{i,j-1}-2\tilde{P}^{Q}_{i,j}\right)-\left[\frac{2}{\Delta X}-\frac{1}{A^n_{i,j}}\right]\tilde{P}^{Q}_{i,j}-G^n_{i,j}+\rho^n_{i,j}\left[1-\frac{I^{n+1}_{i,j}}{I^n_{i,j}}\right] \qquad (4\text{-}28a)$$

and

$$\left[\frac{\theta\varphi(\Delta t)^2 N_{Eu}R_A^2}{N_{S\ell}^2(\Delta R)^2}\right]\tilde{P}^{Q+1}_{i,j-1}+\left[-\frac{2\theta\varphi(\Delta t)^2 N_{Eu}R_A^2}{N_{S\ell}^2(\Delta R)^2}-\frac{2}{\Delta X}\right]\tilde{P}^{Q+1}_{i,j}+\left[\frac{\theta\varphi(\Delta t)^2 N_{Eu}R_A^2}{N_{S\ell}^2(\Delta R)^2}\right]\tilde{P}^{Q+1}_{i,j+1}=$$

$$\left[\frac{\theta\varphi(\Delta t)^2 N_{Eu}R_A^2}{N_{S\ell}^2(\Delta R)^2}\right]\left(\tilde{P}^{Q}_{i,j+1}+\tilde{P}^{Q}_{i,j-1}-2\tilde{P}^{Q}_{i,j}\right)+\frac{2}{\Delta X}\left(\tilde{P}^{Q}_{i,j}-2\tilde{P}^{Q+1/2}_{i,j}\right) \qquad (4\text{-}28b)$$

As with SOR, the ADI method approaches the true $\tilde{P}$ field to a specified accuracy. The superscript Q refers to the Qth iteration.

The R- and Z- momentum equations yield advanced values for V and U.

$$V^{n+1}_{i-1/2,j}=\left\{\left(\rho^n_{i,j}+\rho^n_{i-1,j}\right)V^n_{i-1/2,j}+\left[\frac{2\Delta t}{N_{S\ell}\Delta R}\right]\left[N_{Eu}\varphi\left(\tilde{P}_{i-1,j}-\tilde{P}_{i,j}\right)\right.\right.$$

$$\left.\left.+N_{Eu}(1-\varphi)\left(P^n_{i-1,j}-P^n_{i,j}\right)+B^n_{i-1/2,j}\right]\right\}\Big/\left(\rho^{n+1}_{i,j}+\rho^{n+1}_{i-1,j}\right) \qquad (4\text{-}29)$$

$$U^{n+1}_{i,j-1/2} = \Bigg\{ \left(\rho^{n}_{i,j} + \rho^{n}_{i-1,j}\right) U^{n}_{i,j-1/2} + \left[\frac{2\Delta t}{N_{S\ell}\Delta R}\right]\Big[R_A N_{Eu}\varphi\left(\tilde{P}_{i,j-1} - \tilde{P}_{i,j}\right) + R_A N_{Eu}(1-\varphi)\left(P^{n}_{i,j-1} - P^{n}_{i,j}\right) + D^{n}_{i,j-1/2}\Big]\Bigg\} \Big/ \left(\rho^{n+1}_{i,j} \quad \rho^{n+1}_{i,j-1}\right) \tag{4-30}$$

The total energy equation is written with almost all advanced time terms.

$$\begin{aligned} E^{n+1}_{i,j} = {} & \Bigg[\rho^{n}_{i,j} E^{n}_{i,j} + \left[\frac{\Delta t}{2\Delta R N_{S\ell}}\right]\Bigg\{ \left(\frac{i-2}{2i-3}\right)\left(\rho^{n+1}_{i,j} + \rho^{n+1}_{i-1,j}\right)\left(E^{n}_{i,j} + E^{n}_{i-1,j}\right) V^{n+1}_{i-1/2,j} \\ & - \left(\frac{i-1}{2i-3}\right)\left(\rho^{n+1}_{i,j} + \rho^{n+1}_{i+1,j}\right)\left(E^{n}_{i,j} + E^{n}_{i+1,j}\right) V^{n+1}_{i+1/2,j} \\ & + \frac{1}{2} R_A \Big[\left(\rho^{n+1}_{i,j} + \rho^{n+1}_{i,j-1}\right)\left(E^{n}_{i,j} + E^{n}_{i,j-1}\right) U^{n+1}_{i,j-1/2} - \left(\rho^{n+1}_{i,j} + \rho^{n+1}_{i,j+1}\right)\left(E^{n}_{i,j} + E^{n}_{i,j+1}\right) U^{n+1}_{i,j+1/2} \Big] \Bigg\} \\ & + \left[\frac{\Delta t N_{Eu}}{\Delta R N_{S\ell}}\right]\Bigg\{ \left(\frac{i-2}{2i-3}\right)\left(\tilde{P}_{i,j} + \tilde{P}_{i-1,j} + 2\right) V^{n+1}_{i-1/2,j} - \left(\frac{i-1}{2i-3}\right)\left(\tilde{P}_{i,j} + \tilde{P}_{i+1,j} + 2\right) V^{n+1}_{i+1/2,j} \\ & + \frac{1}{2} R_A \Big[\left(\tilde{P}_{i,j} + \tilde{P}_{i,j-1} + 2\right) U^{n+1}_{i,j-1/2} - \left(\tilde{P}_{i,j} + \tilde{P}_{i,j+1} + 2\right) U^{n+1}_{i,j+1/2} \Big] \Bigg\} \\ & + \left[\frac{\Delta t}{2(\Delta R)^2 N_{S\ell}}\right]\left[\frac{1}{(N_{Re})^{n}_{i,j}}\right]\Bigg\{ 4\left[\frac{\gamma^{n}_{i,j}}{(N_{Pr})^{n}_{i,j}}\right]\Big[\left(\frac{i-1}{2i-3}\right)\left(I^{n}_{i+1,j} - I^{n}_{i,j}\right) - \left(\frac{i-2}{2i-3}\right)\left(I^{n}_{i,j} - I^{n}_{i-1,j}\right) \end{aligned}$$

(equation (4-31) continued on next page)

$$+\left(\frac{i-1}{2i-3}\right)\Bigg[-\left(Q^{n}_{i,j}+Q^{n}_{i+1,j}\right)V^{n+1}_{i+1/2,j}+4V^{n+1}_{i+1/2,j}\left(V^{n+1}_{i+3/2,j}-V^{n+1}_{i+1/2,j}\right)$$

$$+2\left(U^{n+1}_{i+1,j+1/2}U^{n+1}_{i+1,j-1/2}-U^{n+1}_{i,j+1/2}U^{n+1}_{i,j-1/2}\right)+\frac{1}{2}R_A\left(U^{n+1}_{i+1,j+1/2}+U^{n+1}_{i,j+1/2}\right.$$

$$\left.+U^{n+1}_{i+1,j-1/2}+U^{n+1}_{i,j-1/2}\right)\left(V^{n+1}_{i+1/2,j+1}-V^{n+1}_{i+1/2,j-1}\right)\Bigg]-\left(\frac{i-2}{2i-3}\right)\Bigg[-\left(Q^{n}_{i,j}+Q^{n}_{i-1,j}\right)V^{n+1}_{i-1/2,j}$$

$$+4V^{n+1}_{i-1/2,j}\left(V^{n+1}_{i+1/2,j}-V^{n+1}_{i-3/2,j}\right)+2\left(U^{n+1}_{i,j+1/2}U^{n+1}_{i,j-1/2}-U^{n+1}_{i-1,j+1/2}U^{n+1}_{i-1,j-1/2}\right)$$

$$+\frac{1}{2}R_A\left(U^{n+1}_{i,j+1/2}+U^{n+1}_{i-1,j+1/2}+U^{n+1}_{i,j-1/2}+U^{n+1}_{i-1,j-1/2}\right)\left(V^{n+1}_{i-1/2,j+1}-V^{n+1}_{i-1/2,j-1}\right)\Bigg]\Bigg\}$$

$$+\left[\frac{\Delta t}{4(\Delta R)^2 N_{S\ell}}R_A\right]\left[\frac{1}{(N_{Re})^{n}_{i,j}}\right]\Bigg\{4R_A\left[\frac{\gamma^{n}_{i,j}}{(N_{Pr})^{n}_{i,j}}\right]\left(I^{n}_{i,j+1}+I^{n}_{i,j-1}-2I^{n}_{i,j}\right)$$

$$-\left(Q^{n}_{i,j+1}+Q^{n}_{i,j}\right)U^{n+1}_{i-1/2,j+1}+2R_A\Bigg[\left(V^{n+1}_{i+1/2,j+1}V^{n+1}_{i-1/2,j+1}-V^{n+1}_{i+1/2,j}V^{n+1}_{i-1/2,j}\right)$$

$$+2U^{n+1}_{i,j+1/2}\left(U^{n+1}_{i,j+3/2}-U^{n+1}_{i,j-1/2}\right)\Bigg]+\frac{1}{2}\left(V^{n+1}_{i+1,j+1/2}+V^{n+1}_{i,j+1/2}\right.$$

$$\left.+V^{n+1}_{i+1,j-1/2}+V^{n+1}_{i,j-1/2}\right)\left(U^{n+1}_{i+1/2,j+1}-U^{n+1}_{i+3/2,j+1}\right)+\left(Q^{n}_{i,j}+Q^{n}_{i,j-1}\right)U^{n}_{i,j-1/2}$$

(equation (4-31) continued on next page)

$$- 2R_A \left[\left(V^{n+1}_{i+1/2,j} V^{n+1}_{i-1/2,j} - V^{n+1}_{i+1/2,j-1} V^{n+1}_{i-1/2,j-1} \right) + 2U^{n+1}_{i,j-1/2} \left(U^{n+1}_{i,j+1/2} - U^{n+1}_{i,j-3/2} \right) \right.$$

$$\left. - \frac{1}{2} \left(V^{n+1}_{i+1/2,j} + V^{n+1}_{i-1/2,j} + V^{n+1}_{i+1/2,j-1} + V^{n+1}_{i-1/2,j-1} \right) \left(U^{n+1}_{i+1,j-1/2} - U^{n+1}_{i-1,j-1/2} \right) \right] \Bigg\} \Bigg] \Big/ \rho^{n+1}_{i,j} \tag{4-31}$$

The internal energy is simply

$$I^{n+1}_{i,j} = E^{n+1}_{i,j} - \frac{1}{8} \left[\left(V^{n+1}_{i+1/2,j} + V^{n+1}_{i-1/2,j} \right)^2 + \left(U^{n+1}_{i,j+1/2} + U^{n+1}_{i,j-1/2} \right)^2 \right] \tag{4-32}$$

The above equations (4-22) to (4-32) plus equation (4-15) are sufficient for single fluid problems. If species diffusion and reaction are required, the conservation of component mass equation must be included. This is finite differenced as,

$$N_{S\ell} \left[\frac{(\rho_K)^{n+1}_{i,j} - (\rho_K)^{n}_{i,j}}{\Delta t} \right] = - \left[\frac{\psi}{\Delta R} \right] \left\{ \left(\frac{i-1}{2i-3} \right) \left[(\rho_K)^{n+1}_{i+1,j} + (\rho_K)^{n+1}_{i,j} \right] V^{n+1}_{i+1/2,j} \right.$$

$$- \left(\frac{i-2}{2i-3} \right) \left[(\rho_K)^{n+1}_{i,j} + (\rho_K)^{n+1}_{i-1,j} \right] V^{n+1}_{i-1/2,j} + \frac{1}{2} R_A \left[(\rho_K)^{n+1}_{i,j+1} + (\rho_K)^{n+1}_{i,j} \right] U^{n+1}_{i,j+1/2}$$

$$\left. - \frac{1}{2} R_A \left[(\rho_K)^{n+1}_{i,j} + (\rho_K)^{n+1}_{i,j-1} \right] U^{n+1}_{i,j-1/2} \right\}$$

(equation (4-33) continued on next page)

$$
- \left[\frac{1-\psi}{\Delta R}\right]\left\{\left(\frac{i-1}{2i-3}\right)\left[(\rho_K)^n_{i+1,j} + (\rho_K)^n_{i,j}\right] V^{n+1}_{i+1/2,j} - \left(\frac{i-2}{2i-3}\right)\left[(\rho_K)^n_{i,j} + (\rho_K)^n_{i-1,j}\right] V^{n+1}_{i-1/2,j}\right.
$$

$$
\left. + \frac{1}{2} R_A\left[(\rho_K)^n_{i,j+1} + (\rho_K)^n_{i,j}\right] U^{n+1}_{i,j+1/2} - \frac{1}{2} R_A\left[(\rho_K)^n_{i,j} + (\rho_K)^n_{i,j-1}\right] U^{n+1}_{i,j-1/2}\right.
$$

$$
+ \left[\frac{4M_K}{(\Delta R)^2}\right]\left\{\left(\frac{i-1}{2i-3}\right)\left[(M_{R-K})^n_{i,j} \frac{\left(C^n_{i+1,j} + C^n_{i,j}\right)\left(X^n_{i+1,j} - X_{i,j}\right)}{\left(M^n_{i+1,j} + M^n_{i,j}\right)\left[(N_{Re}N_{Sc})^n_{i+1,j} + (N_{Re}N_{Sc})^n_{i,j}\right]}\right]\right.
$$

$$
+ \left[\frac{4M_K}{(\Delta R)^2}\right]\left\{\left(\frac{i-1}{2i-3}\right) \frac{\left(C^n_{i+1,j} + C^n_{i,j}\right)\left[(M_{R-K})^n_{i+1,j} + (M_{R-K})^n_{i,j}\right]\left[(X_K)^n_{i+1,j} - (X_K)^n_{i,j}\right]}{\left(M^n_{i+1,j} + M^n_{i,j}\right)\left[(N_{Re})^n_{i+1,j} + (N_{Re})^n_{i,j}\right]\left[(N_{Sc})^n_{i+1,j} + (N_{Sc})^n_{i,j}\right]}\right.
$$

$$
\left. - \left(\frac{i-2}{2i-3}\right)\left[\frac{\left(C^n_{i,j} + C^n_{i-1,j}\right)\left[(M_{R-K})^n_{i,j} + (M_{R-K})^n_{i-1,j}\right]\left[(X_K)^n_{i,j} - (X_K)^n_{i-1,j}\right]}{\left(M^n_{i,j} + M^n_{i-1,j}\right)\left[(N_{Re})^n_{i,j} + (N_{Re})^n_{i-1,j}\right]\left[(N_{Sc})^n_{i,j} + (N_{Sc})^n_{i-1,j}\right]}\right]
$$

$$
+ \frac{1}{2} R^2_A\left[\frac{\left(C^n_{i,j+1} + C^n_{i,j}\right)\left[(M_{R-K})^n_{i,j+1} + (M_{R-K})^n_{i,j}\right]\left[(X_K)^n_{i,j+1} - (X_K)^n_{i,j}\right]}{\left(M^n_{i,j+1} + M^n_{i,j}\right)\left[(N_{Re})^n_{i,j+1} + (N_{Re})^n_{i,j}\right]\left[(N_{Sc})^n_{i,j+1} + (N_{Sc})^n_{i,j}\right]}\right.
$$

$$
\left. - \frac{1}{2} R^2_A\left[\frac{\left(C^n_{i,j} + C^n_{i,j-1}\right)\left[(M_{R-K})^n_{i,j} + (M_{R-K})^n_{i,j-1}\right]\left[(X_K)^n_{i,j} - (X_K)^n_{i,j-1}\right]}{\left(M^n_{i,j} + M^n_{i,j-1}\right)\left[(N_{Re})^n_{i,j} + (N_{Re})^n_{i,j-1}\right]\left[(N_{Sc})^n_{i,j} + (N_{Sc})^n_{i,j-1}\right]}\right]\right\}
$$

$$
+ N_{S\ell} \sum_{y=1}^{Y}\left[\frac{k'}{Y} \rho^A_K \rho^B_L\right]^{n+(y-1)/Y}_{i,j} + (\beta_{MK})^n_{i,j} \qquad (4\text{-}33)
$$

The binary form is used in this study. Only the molecular diffusion term differs in the multicomponent version, and its form is easily seen from equation (4-33). The convective fluxes are written both implicitly and explicitly with a proportioning coefficient ψ. As with θ and φ, $0 \leq \psi \leq 1$. The diffusion term is written explicitly, and the reaction term implicitly. This choice of implicit vs. explicit finite differencing evolved from numerical experimentation performed herein.

The implicit nature of the equations requires an implicit scheme, and the ADI method is used. The equation is solved in a two step fashion, the first step presuming diffusion without chemical reaction and the second step presuming reaction only. To assure convergence, the time step Δt is subdivided into smaller increments, $\Delta t'$, for the diffusion calculation. The reaction step subdivides the increment $\Delta t'$ into even smaller time steps $\Delta t''$. This process is necessary because the characteristic time of reaction kinetics is often considerably shorter than the characteristic time of fluid motion. A suitable Δt for fluid flow may be many times too large for chemical reaction, resulting in reaction overshoot, errors, and usually numerical instability. Equation (4-33) is written over the time increment Δt, but it is used in the computer program over the smaller increment $\Delta t'$. Thus the superscript notation shown is not exactly correct for an increment $\Delta t'$. In practice the explicit terms are calculated, then the diffusion without reaction is calculated over $\Delta t'$ for each specie. Next the reaction is presumed to take place without diffusion in a series of time steps $\Delta t'' = \Delta t'/Y$. This reaction step is done Y times, each

step using the results of the previous reaction. Then diffusion is again calculated over the next $\Delta t'$ using specie densities computed from the last $\Delta t'$ in the implicit portion of the equation. The process is repeated until the true time step Δt is bridged.

The ADI formulation used in the diffusion calculation across increment $\Delta t'$ is

$$\left[-\left(\frac{i-2}{2i-3}\right)V^{n+1}_{i-1/2,j}\right](\rho_K)^{n+1/2}_{i-1,j} + \left[\left(\frac{i-1}{2i-3}\right)V^{n+1}_{i+1/2,j} - \left(\frac{i-2}{2i-3}\right)V^{n+1}_{i-1/2,j} + \frac{2N_{S\ell}\Delta R}{\Delta t'\psi}\right](\rho_K)^{n+1/2}_{i,j} + \left[\left(\frac{i-1}{2i-3}\right)V^{n+1}_{i+1/2,j}\right](\rho_K)^{n+1/2}_{i+1,j}$$

$$= \left[\frac{1}{2}R_A U^{n+1}_{i,j-1/2}\right](\rho_K)^{n}_{i,j-1} - \left[\frac{1}{2}R_A U^{n+1}_{i,j+1/2} - \frac{1}{2}R_A U^{n+1}_{i,j-1/2} - \frac{2N_{S\ell}\Delta R}{\Delta t\psi}\right](\rho_K)^{n}_{i,j} - \left[\frac{1}{2}R_A U^{n+1}_{i,j+1/2}\right](\rho_K)^{n}_{i,j+1} + (S_K)^{n}_{i,j} \qquad (4\text{-}34a)$$

$$\left[U^{n+1}_{i,j-1/2}\right](\rho_K)^{n+1}_{i,j-1} + \left[U^{n+1}_{i,j-1/2} - U^{n+1}_{i,j+1/2} - \frac{4N_{S\ell}\Delta R}{\Delta t'\psi R_A}\right](\rho_K)^{n+1}_{i,j} + \left[-U^{n+1}_{i,j+1/2}\right](\rho_K)^{n+1}_{i,j+1/2} = \left[U^{n+1}_{i,j-1/2}\right](\rho_K)^{n}_{i,j-1} + \left[U^{n+1}_{i,j-1/2} - U^{n+1}_{i,j+1/2} + \frac{4N_{S\ell}\Delta R}{\Delta t'\psi R_A}\right](\rho_K)^{n}_{i,j} + \left[-U^{n+1}_{i,j+1/2}\right](\rho_K)^{n}_{i,j+1} - 2\left[\frac{4N_{S\ell}\Delta R}{\Delta t'\psi R_A}\right](\rho_K)^{n+1/2}_{i,j} \qquad (4\text{-}34b)$$

The explicit terms in equation (4-33), including the truncation error correction β_{MK}, are multiplied by $\Delta R/\psi$ and collected under the term S_K in equation (4-34a). The chemical reaction equation is simply

$$(\rho_K)_{i,j}^{n+y/Y} = (\rho_K)^{n+(y-1)/Y} + \Delta t'' k' \left(\rho_K^A\right)_{i,j}^{n+(y-1)/Y} \left(\rho_L^B\right)_{i,j}^{n+(y-1)/Y} \qquad (4\text{-}35)$$

As mentioned before, this is performed Y times in each increment $\Delta t'$.

The full set of finite difference equations for compressible flows with specie diffusion and chemical reaction has been presented. The next task is to examine the application of boundary conditions to the ICE computational scheme.

D. Boundary Conditions

A typical cell and the computational mesh was shown in figures 1 and 2. The ICE system is especially convenient in applying boundary conditions. In all problems considered here, boundaries are aligned with cell walls. That is, there are no partial cells with boundaries passing through the cell interior. As mentioned earlier, the grid has four types of boundaries: centerline, wall, input, and output. There are also three types of boundary conditions. They are based on assumptions concerning the boundary value of a variable itself, or on its first or second spatial derivatives.

The centerline boundary is symmetrical, presuming no radial flux is possible across the centerline of a tube with flow independent of angular position. Assume the cell i,j is within the computational grid and the wall at $i - 1/2$ lies on the centerline. Then

$$V_{i-1/2,j} = 0 \qquad (4\text{-}36)$$

All other variables, such as U, are symmetrical.

$$U_{i-1,j-1/2} = U_{i,j-1/2} \qquad (4\text{-}37)$$

Nonzero variables are not defined on the centerline $i - 1/2$, making it unnecessary to write particular equations obtained by evaluating limits as $1/R \to 0$.

Wall boundary conditions are usually straightforward. A no-slip wall means the axial velocity is zero at the wall. But U is not defined at the wall. Therefore, an average must be formed at the wall which is zero, and this prescribes the value of U in the artificial row of cells which impresses boundary conditions. If the cell i,j has its wall at $i + 1/2$ aligned on the tube wall,

$$U_{i+1,j-1/2} = - U_{i,j-1/2} \tag{4-38}$$

If the wall is full slip, the axial velocity radial gradient is zero, and

$$U_{i+1,j-1/2} = U_{i,j-1/2} \tag{4-39}$$

For an impermeable wall,

$$V_{i+1/2,j} = 0 \tag{4-40}$$

If the wall is reflective,

$$P_{i+1,j} = P_{i,j} \tag{4-41}$$

$$V_{i+3/2,j} = - V_{i-1/2,j} \tag{4-42}$$

Other assumptions may be made to evaluate $V_{i+3/2,j}$. These are based on the velocity divergence, Q_{ij} which is presumed to be reflective if the density is reflective. Then

$$Q_{i+1,j} = Q_{i,j} \tag{4-43}$$

This can be solved for $V_{i+3/2,j}$ if the wall permeability and slip is

specified. This condition also removes a gradient term at the wall in the energy equation. For incompressible flows all Q's are zero. If the wall is insulated,

$$E_{i+1,j} = E_{i,j} \tag{4-44}$$

$$I_{i+1,j} = I_{i,j} \tag{4-45}$$

An especially superior boundary condition can be found on $\tilde{P}$ and P at the wall if all velocities are known. If the wall is impermeable, $\partial V/\partial t$ is zero, and this may be used with the finite difference form of the radial momentum equation to give $\tilde{P}$ and P. The coefficient φ is arbitrary and may be set to either 1 and 0, leading to

$$\tilde{P}_{i+1,j} = \tilde{P}_{i,j} + B^{n}_{i+1/2,j}/N_{Eu} \tag{4-46}$$

$$P_{i+1,j} = P_{i,j} + B^{n}_{i+1/2,j}/N_{Eu} \tag{4-47}$$

Care must be taken that the equation of state is satisfied at the wall. Thus if P and ρ are specified, I is also known.

The upstream boundary condition is usually a specified input. Again presume the cell i,j lies in the computational grid with the cell wall at $j - 1/2$ coincident with the input boundary. For all problems run herein the radial velocity is assumed zero. Since V is not defined on the input boundary, an average is formed of straddling axial values, and

$$V_{i,j-1/2} = V_{i,j+1/2} \tag{4-48}$$

If P and I are specified, ρ is known, $U_{i,j-1/2}$ is calculated, and $U_{i,j-3/2}$ is found by an assumption on Q at the input. Usually this

is not critical. The input total energy may be calculated after $U_{i,j-1/2}$ is known. Conversely, if the input velocity is prescribed, the axial momentum equation will specify the pressure gradient at the input and, for arbitrary φ,

$$\tilde{P}_{i,j-1} = \tilde{P}_{i,j} - D_{i,j-1/2}/(R_A N_{Eu}) \tag{4-49}$$

A similar equation is written for $P_{i,j-1}$. Then ρ may be found from the equation of state.

The least resolved boundary conditions are at the output. Often the pressure is known or may be calculated as a function of velocity, but other variables are not known. LASL often uses the concept of a continuative output. This is a boundary condition which does not propagate signals far upstream, since all gradients are set equal to zero. The problem may be distorted in the region near the continuative output boundary. Paris (41) presumes the output boundary is so far downstream that at steady state a known flow, such as parabolic Poisuille flow, exists at the output. The output boundary is moved farther downstream in a series of numerical experiments until no further change occurs in the solution. But several problems in the present study have outlet boundaries that are not far downstream. It is thus not possible to use the concepts of Paris in these cases.

The continuative output is desirable for many of the prescribed input flows, but the possible distortion near the output boundary is unwanted. It is found that a reasonable output flow results when the pressure is fixed and the remaining variables are assumed to have constant first derivatives at, or near the output. Thus the second derivatives are zero.

$$\frac{U_{i,j+3/2} - 2U_{i,j+1/2} + U_{i,j-1/2}}{(\Delta Z)^2} = 0 \qquad (4\text{-}50)$$

leading to

$$U_{i,j+3/2} = 2U_{i,j+1/2} - U_{i,j-1/2} \qquad (4\text{-}51)$$

and

$$\rho_{i,j+1} = 2\rho_{i,j} - \rho_{i,j-1} \qquad (4\text{-}52)$$

and so forth.

Note that the (4-52) condition presumes the second derivative is zero at the center of the last cell upstream of the output boundary, rather than at the boundary itself. If a zero second derivative of any cell-centered variable is applied at the output boundary $j + 1/2$ to find a value at $j + 1$, variables at both $j + 2$ and $j + 1$ arise. Thus the number of unknown values is not reduced. The definition (4-52) applied to all cell center variables uses information at two upstream locations and produces a smooth output. Heuristic reasoning suggests that if a zero axial second derivative is applied a half cell upstream of the true output boundary, it is also effectively applicable at the boundary. Conversely, it must be recognized that a slight distortion may be introduced upstream of the output boundary.

The boundary conditions on specie densities offer no new problems. Symmetry still applies at the tube centerline, the wall radial gradient is zero to prevent mass transfer through the impermeable wall, the input is specified, and the output is continuous in the sense that the second axial derivative of the densities is assumed zero near the output boundary. It should be noted that for some variables a reverse

flow from outside of the tube into the downstream end of the tube would cause no real problem in the computation. But for specie densities such a flow reversal would be disasterous since the degree of mixing and reaction outside of the tube is entirely unknown.

This concludes the general description of boundary conditions. The conditions used for each problem will be discussed in the results section. The same finite-differenced conservation equations apply over the interior cells for all problems. A specific problem can be calculated only by impressing suitable boundary conditions. If the boundary conditions are unrealistic, so is the resulting problem. Thus, the application of boundary conditions must be done with extreme care.

E. Computer Solution of the Equations

An examination of the finite difference equations show a large number of implicit terms (assuming φ and $\theta \neq 0$). This favors stability and gives confidence in the time dependent aspects of the problem. It also can increase certain truncation errors and can lead to long numerical iterations. The implicit nature of the equations is needed for fluid flows at low speeds. This is because slow flows become incompressible in behavior and require knowledge of the entire pressure field. This knowledge is propagated through the grid by the implicit solution process.

The basic ICE method requires iteration only for $\tilde{P}$. All other variables may be calculated in an explicit manner. If the expanded definition of the hybrid function as presented by this author is used, an iteration on I^{n+1} is required. Since the inner $\tilde{P}$ iteration is

nested within the outer I^{n+1} iteration, the expanded algorithm should be avoided if possible. Fortunately, this usually can be done. The iteration required by the implicit convective fluxes in the species equation is not nested with the other two iterations, and it does not incur a severe computer time penalty.

An outline of the algorithm is shown in figure 3. After an initialization sequence the main time loop is entered. Explicit terms including all truncation corrections and Q, B, D, and G are calculated. If the outer iteration on I^{n+1} is required, I^{n+1} is initially set equal to I^{n}. The $\tilde{P}$ iteration commences using either equations (4-27) or (4-28a,b). Upon convergence to a suitably small change, the advanced density ρ^{n+1} is computed from equation (4-18). Next V^{n+1} and U^{n+1} may be calculated using equations (4-29) and (4-30). At this point the continuity equation is checked to assure that it is satisfied to at least 3.5×10^{-3}, a number which LASL found a minimal for satisfactory accuracy and stability (55). If this test is exceeded for any cell, the program performs more $\tilde{P}$ iterations and recalculates ρ^{n+1}. When the continuity test is satisfied, E^{n+1} and I^{n+1} are calculated using equations (4-31) and (4-32). If an outer iteration is required on I^{n+1}, the program may return to the $\tilde{P}$ iteration with the improved value of I^{n+1}, or the program flow may proceed. If the fluid is designated as incompressible, the speed of sound is set to a large value (10^{16}) and the energy calculation is bypassed.

If the fluid is multicomponent, the species equation is solved by computing the explicit term S_K, and iterating on $(\rho_K)^{n+1}$ with a number of reaction steps after each iteration. After convergence the

advanced pressure P^{n+1} is computed from the equation of state (4-15) and the time cycle is completed.

Further details may be found in Appendices A, B, and C. Appendix A discusses the SOR and ADI schemes for solving the $\tilde{P}$ equation, Appendix B proposes a sequence for solving the reaction equations with strong temperature effects, and Appendix C documents the current numerical program which is still in the research phase.

This concludes the development of the finite difference equation and the outline of their solution. The next chapter discusses stability analyses and the truncation error corrections, β.

Chapter 5

STABILITY AND TRUNCATION ERROR ANALYSIS

The problem of numerical stability was briefly discussed in the introduction. This chapter examines the matter of stability in more detail and presents several methods which can predict stable bounds on time and space increment sizes. Also, the Courant-Friedricks-Lewy necessary condition for stability has been adequately covered in the introduction. The approach of Cheng mentioned in the introduction is similar to that of Hirt, but Hirt's analysis is preferred.

A simple equation containing both a convection and a diffusion term is used to illustrate the methods. The analysis by Hirt is applied in more detail to simple conservation equations. It is used in this thesis with the two dimensional ICE equations to generate terms which correct for truncation errors that produce instability and loss of accuracy. The corrections are presented and finite differenced. Besides the original papers, the excellent presentation in the text by Roache (45) is used heavily in this chapter.

A. Basic Concepts

It is more correct to say that the numerical fluid dynamicist is faced with the problem of instability rather than the problem of stability. A computation may be stable and incorrect due to unwanted

numerical diffusion, but an unstable computation is usually catastrophic. Stability analyses are difficult. Most equations of interest are nonlinear, whereas most analyses are linear, and their application to multidimensional equations which are not simple in form involves ponderous algebra. But linear analyses do provide some insight into the behavior of the equations, so such analyses are worthwhile in assisting the fluid dynamicist toward producing a numerical solution which approaches the physics of the flows.

Two types of numerical instability are found in computations: static and dynamic. Consider a variable W which is distributed over a coordinate Z at a time t. Presume that this distribution represents a steady state solution of $W(Z,t)$. Now perturb the solution in a point to point manner along Z such that the perturbation oscillates, producing a saw-tooth error curve over Z. Three types of dynamic behavior can occur at later time steps. The perturbations can die out, then the finite-difference solution is stable. The perturbations can grow monotonically, the deflection increasing at each time step so that a positive perturbation remains positive and grows larger. This is static instability. The perturbations can not only grow but also change sign, flip-flopping around the true steady state solution. This is defined as dynamic instability. One instance of dynamic instability occurs when a normally stable explicit scheme is run at too large a time step, the large Δt causing drastic overshoot on all deflections. Both static and dynamic instabilities were encountered in the present study.

The analyses of the next section will be illustrated using the following simple partial differential equation in one spatial dimension with constant coefficients, U and ν.

$$\frac{\partial W}{\partial t} = - U \frac{\partial W}{\partial Z} + \nu \frac{\partial^2 W}{\partial Z^2} \tag{5-1}$$

This equation has a convective and diffusive term on the right hand side and thus simulates the form of the conservation equations. The finite difference form of (5-1) will be taken as

$$\frac{W_j^{n+1} - W_j^n}{\Delta t} = - U \frac{W_{j+1}^n - W_{j-1}^n}{2\Delta Z} + \nu \frac{W_{j+1}^n - 2W_j^n + W_{j-1}^n}{(\Delta Z)^2} \tag{5-2}$$

The time difference is forward, the others are centered. The convecting velocity U is constant in time and space.

B. Linear Stability Analyses

a) Method of Positive Coefficients

This is a simple criteria where the equation is rearranged and the coefficients of each W term examined. If all coefficients are positive, the equation is stable, according to Forsythe and Wasow (18). Rearranging (5-2) gives

$$[1]W_j^{n+1} = \left[\frac{\Delta t}{\Delta Z}\left(-\frac{U}{2} + \frac{\nu}{\Delta Z}\right)\right] W_{j+1}^n + \left[1 - 2\nu \frac{\Delta t}{(\Delta Z)^2}\right] W_j^n + \left[\frac{\Delta t}{\Delta Z}\left(\frac{U}{2} + \frac{\nu}{\Delta Z}\right)\right] W_{j-1}^n \tag{5-3}$$

Note the appearance of a Courant number, $U\Delta t/\Delta Z$, in two of the coefficients. The conditions which lead to stability are found from the coefficients in the brackets. For each W these are

$$W^n_{j+1}: \qquad U \leq \frac{2\nu}{\Delta Z} \tag{5-4}$$

$$W^n_{j} : \qquad \Delta t \leq \frac{(\Delta Z)^2}{2\nu} \tag{5-5}$$

$$W^n_{j-1}: \qquad U \geq -\frac{2\nu}{\Delta Z} \tag{5-6}$$

An equivalent equation to (5-4) and (5-6) is

$$U^2 \leq \frac{4\nu^2}{(\Delta Z)^2} \tag{5-7}$$

combining this with (5-5) leads to

$$\Delta t \leq \frac{2\nu}{U^2} \tag{5-8}$$

Furthermore, if (5-4) is rearranged to give

$$\frac{\Delta Z}{2\nu} < \frac{1}{U} \tag{5-9}$$

and this is used in (5-5), the result is

$$U \frac{\Delta t}{\Delta Z} < 1 \tag{5-10}$$

which is the Courant condition. Equations (5-5) and (5-8) describe the restrictions on Δt, while (5-4) and (5-6) show the limits of $|U|$ as a function of ν and ΔZ, independent of Δt. A large diffusion coefficient or small mesh is necessary to permit a usable velocity range. A computation for an inviscid fluid ($\nu = 0$) would be unstable. This has been verified by Cheng (11) and others for this finite difference scheme.

b) Method of Discrete Perturbation

This method provides some insight to stability phenomena by presuming the solution is at steady state, then locally imposing a

perturbation of W at a point and examining the requirements which prevent the perturbed value, W', from becoming unbounded. If W' is applied at point (j), and the finite difference equation is written at point (j + 1) to generate information in the convection term,

$$\frac{W_{j+1}^{n+1} - W_{j+1}^{n}}{\Delta t} = - U \frac{W_{j+2}^{n} - \left(W_{j}^{n} + W'\right)}{2\Delta Z} + \nu \frac{W_{j+2}^{n} - 2W_{j}^{n} + \left(W_{j}^{n} + W'\right)}{(\Delta Z)^2} \tag{5-11}$$

To isolate the behavior of the perturbation, assume the steady state solution gives $W_j^n = 0$ for all j. Then

$$\frac{W_{j+1}^{n+1}}{\Delta t} = U \frac{W'}{2\Delta Z} + \frac{W'}{(\Delta Z)^2} \tag{5-12}$$

For stability to be assured, the response of the system to the normalized perturbation must be

$$\left| \frac{W_{j+1}^{n+1}}{W'} \right| \leq 1 \tag{5-13}$$

This means

$$- 1 \leq U \frac{\Delta t}{2\Delta Z} + \nu \frac{\Delta t}{(\Delta Z)^2} \leq 1 \tag{5-14}$$

The right inequality specifies static instability bounds.

$$\Delta t \leq \frac{\Delta Z}{\frac{U}{2} + \frac{\nu}{\Delta Z}} \tag{5-15}$$

Positive Δt occurs if $U > - 2\nu/\Delta Z$. This is the constraint (5-6) which arose in the positive coefficient analysis. The left inequality in (5-14) specifies dynamic instability bounds, and it is also satisfied by (5-6).

If the entire analysis is repeated at $(j - 1)$, the restrictions which arise are

$$-1 \leq -U\frac{\Delta t}{2\Delta Z} + \nu\frac{\Delta t}{(\Delta Z)^2} \leq 1 \tag{5-16}$$

or

$$\Delta t \leq \frac{\Delta Z}{-\frac{U}{2} + \frac{\nu}{\Delta Z}} \tag{5-17}$$

This is satisfactory if $U \leq 2\nu/\Delta Z$, which is relation (5-4). Thus the analysis gives the same restrictions on $|U|$ as the positive coefficient analysis.

Equation (5-15) may be examined under the restriction $|U| \leq 2\nu/\Delta Z$ It is seen that the minimum upper limit on Δt occurs when $U = 2\nu/\Delta Z$, and the restriction (5-5) is the result. Hence the discrete perturbation method gives identical results as the positive coefficient method. Equation (5-5) can also be produced by the discrete perturbation method using two other means; a zero overshoot assumption and an analysis of perturbations that are oscillatory along Z rather than located at one point. These alternate approaches may be found in Roache (45) or the original papers.

c) Karplus' Method of Electric Circuit Stability

A method that is simple to use is the electric circuit analog of Karplus (34). He noted that the current distribution of a network of electrical resistors arranged in a regular pattern could be written as a finite difference equation. Conversely, the finite difference equation could be presumed to have an electric circuit analog. Then

concepts of circuit theory that deal with electrical instability can be applied to finite difference equation instability.

Kirchoff's voltage law expresses the current of loop n, j in terms of currents in adjacent loops.

$$R_1\left(i^n_{j+1} - i^n_j\right) + R_2\left(i^{n+1}_j - i^n_j\right) + R_3\left(i^n_{j-1} - i^n_j\right) + R_4\left(i^{n-1}_j - i^n_j\right) = 0 \tag{5-18}$$

The notation here is conventional electric theory notation, R being resistance and i being current (not to be confused with the index notation of the fluid dynamic equations). The resistance network is stable if a current in a loop dies out after excitation is removed.

Application to finite difference equations is simple. Arrange the equations in the form of (5-15), where j is a bounded space coordinate. If all coefficients (corresponding to R_1, R_2, etc.) are positive, the equation is stable. If some coefficients are negative, the equation is stable if the algebraic sum of all the coefficients is negative. Rearranging equation (5-2) and adding and subtracting the term W^n_j leads to

$$(2\nu - U\Delta Z)\left(W^n_{j+1} - W^n_j\right) + (2\nu + U\Delta Z)\left(W^n_{j-1} - W^n_j\right) - \left(\frac{2(\Delta Z)^2}{\Delta t}\right)\left(W^{n+1}_j - W^n_j\right) = 0 \tag{5-19}$$

The first and second coefficients are positive only in the range proscribed by inequalities (5-4) and (5-6). However, the third coefficient is always negative. Invoking Karplus' second rule for stability,

$$(2\nu - U\Delta Z) + (2\nu + U\Delta Z) - \frac{2(\Delta Z)^2}{\Delta t} < 0 \tag{5-20}$$

which leads to (5-5). Thus the method of Karplus gives identical results to the other methods presented thus far.

d) Method of Von Neumann

This method was mentioned in the introduction and reference (40) was cited. It is the most widely used and has been expanded and modified by many researchers. The method presumes that the solution to the linear partial differential equation is written as an infinite Fourier expansion. The growth or decay of a typical component is studied to determine stability bounds.

Again start with equation (5-2). The Fourier component is

$$W_j^n = S^n e^{ik_z j\Delta Z} = S^n e^{ij\tau} \tag{5-21}$$

where the wave number k_z has been related to a phase angle τ. The quantity i is the square root of minus one. Substituting this into equation (5-2) gives

$$S^{n+1}e^{ij\tau} = S^n e^{ij\tau} - \frac{U\Delta t}{2\Delta Z}\left(S^n e^{i(j+1)\tau} - S^n e^{i(j-1)\tau}\right)$$

$$+ \nu \frac{\Delta t}{(\Delta Z)^2}\left(S^n e^{i(j+1)\tau} - 2S^n e^{ij\tau} + S^n e^{i(j-1)\tau}\right) \tag{5-22}$$

Some manipulation and trignometric substitution gives

$$S^{n+1} = \left[1 - 2\frac{\nu t}{(\Delta Z)^2}(1 - \cos\tau) - iU\frac{\Delta t}{\Delta Z}\sin\tau\right]S^n = HS^n \tag{5-23}$$

H is a complex amplification factor whose modulus $|H|$ must be ≤ 1 for stability. If this restriction is applied, conditions (5-4) and (5-6)

again result. Examination of (5-23) indicates that (5-5) applies, and also that $U \frac{\Delta t}{\Delta Z} < 1$, which is the Courant condition.

As the problem dimensionality and the number of time level increases, the mathematics becomes more complex. The amplification factor H is then a matrix whose eigenvalues must be ≤ 1 if stability is to be possible. Burstein (9) illustrates the problem for a realistic set of unsteady equations in two coordinates.

e) Method of Hirt

Hirt (29) has provided a heuristic approach for analyzing stability. The Taylor series for terms in the finite difference equation (5-2) are written. For example,

$$W_j^{n+1} = W_j^n + \Delta t \left.\frac{\partial W}{\partial t}\right|_j^n + \frac{(\Delta t)^2}{2} \left.\frac{\partial^2 W}{\partial t^2}\right|_j^n + O[(\Delta t)^3] \qquad (5\text{-}24)$$

$$W_{j+1}^n = W_j^n + \Delta Z \left.\frac{\partial W}{\partial Z}\right|_j^n + \frac{(\Delta Z)^2}{2} \left.\frac{\partial^2 W}{\partial Z^2}\right|_j^n + O[(\Delta Z)^3] \qquad (5\text{-}25)$$

$$W_{j-1}^n = W_j^n - \Delta Z \left.\frac{\partial W}{\partial Z}\right|_j^n + \frac{(\Delta Z)^2}{2} \left.\frac{\partial^2 W}{\partial Z^2}\right|_j^n + O[(\Delta Z)^3] \qquad (5\text{-}26)$$

The time difference in (5-2) is contained in (5-24), the convection difference is gotten by (5-25) minus (5-26), and the diffusive second difference by (5-25) plus (5-26).

If these series are substituted into (5-2) and indices are dropped, the result is

$$\frac{\partial W}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 W}{\partial t^2} + O[(\Delta t)^2] = -U \frac{\partial W}{\partial Z} + \nu \frac{\partial^2 W}{\partial Z^2} + O[(\Delta Z)^3] \qquad (5\text{-}27)$$

Omitting the higher order terms and rearranging gives

$$\left(\frac{\Delta t}{2\nu}\right) \frac{\partial^2 W}{\partial t^2} - \frac{\partial^2 W}{\partial Z^2} + \frac{1}{\nu} \frac{\partial W}{\partial t} + \frac{U}{\nu} \frac{\partial W}{\partial Z} = 0 \qquad (5\text{-}28)$$

This is a hyperbolic equation with characteristics $\pm\sqrt{\Delta t/2\nu}$ which mark off a domain of dependence as discussed in the introduction. The difference equation also has a domain of dependence since data W_j^n at a point (j) is propagated over the time increment Δt to the neighboring spatial point $(j \pm 1)$. Thus the domain of dependence of the difference equation is delineated by lines of slope $\pm\Delta t/\Delta Z$. This domain of dependence must contain that of the partial differential equation according to Courant, Friedricks, and Lewy as discussed in the introduction of this thesis. Then

$$\frac{\Delta t}{\Delta Z} \leq \sqrt{\frac{\Delta t}{2\nu}} \qquad (5\text{-}29)$$

which is the same as restriction (5-5).

Another condition of stability is found by differentiating the original partial differential equation with respect to time and reversing the order of differentiation.

$$\frac{\partial}{\partial t}\left(\frac{\partial W}{\partial t}\right) = \frac{\partial^2 W}{\partial t^2} = -U \frac{\partial}{\partial Z}\left(\frac{\partial W}{\partial t}\right) + \nu \frac{\partial^2}{\partial Z^2}\left(\frac{\partial W}{\partial t}\right) \qquad (5\text{-}30)$$

Now the original equation is substituted into the time differentials on the right hand side, and the resulting spatial differentials are expanded.

$$\frac{\partial^2 W}{\partial t^2} = U^2 \frac{\partial^2 W}{\partial Z^2} - 2U\nu \frac{\partial^3 W}{\partial Z^3} + \nu^2 \frac{\partial^4 W}{\partial Z^4} \qquad (5\text{-}31)$$

This is substituted in (5-28), and the third and fourth derivatives

dropped because (1) they are usually small, and (2) the second spatial derivative is the one associated with diffusive damping (or driving).

Rearrangement of the resulting equation gives

$$\frac{\partial W}{\partial t} = - U \frac{\partial W}{\partial Z} + \left(\nu - U^2 \frac{\Delta t}{2} \right) \frac{\partial^2 W}{\partial Z^2} \tag{5-32}$$

A positive diffusion coefficient smears a perturbation in W. If the diffusion coefficient is negative (physically impossible but not mathematically impossible), the perturbation would concentrate and grow. Thus for stability to be assured,

$$\nu - U^2 \frac{\Delta t}{2} \geq 0 \tag{5-33a}$$

or

$$\Delta t \leq \frac{2\nu}{U^2} \tag{5-33b}$$

This result is the same as (5-8).

Although constraints (5-4), (5-5), and (5-6) may be combined to give (5-33b), the converse action may not be taken. There is no way to extract (5-4) and (5-6) from (5-5) and (5-33b), unless the Courant condition (5-10) is used. The first restriction on Δt also used the "domain of dependence" concept.

f) Application of the Various Methods

As the preceding developments showed, all methods gave the same results except that Hirt's analysis did not yield restrictions on $|U|$ without assistance from the Courant condition. The example chosen to illustrate the methods is simple. It can not be concluded that all the methods would give identical results for a complicated equation.

The method of positive coefficients is simple to use but can potentially omit some restrictions. The perturbation method becomes quite ponderous with two dimensional equations. The electrical circuit analog by Karplus is relatively easy to use. Roache mentions an ambiguity in the method but does not explain what this might be. Ghia, Torda, and Lavan (22) used both the Karplus and Von Neumann analyses to determine the stability limits of equations describing steady coaxial flows. The results were identical and the Karplus method was claimed easier to use.

The method of Von Neumann is probably the most widely used because it is well grounded mathematically and conceptually clear. However, finding the eigenvalues of the amplification factor can involve extensive computation. It is also a linear analysis and strictly applicable to Cartesian coordinate systems.

Hirt's analysis of truncation errors becomes ponderous with complex equations. It has predicted regions of instability with success. More important, it suggests a means for removing some of them, which is shown later.

Application of these methods to non-linear equations is done by assuming that the equations are locally linear over small time-space increments. Then stability becomes a point-to-point matter. Mathematical bases for non-linear analyses are lacking. Of the listed methods, those of Karplus and Hirt have the least restrictions, hence the potentially widest applicability. It can be expected that these two methods will be used more frequently in the future.

C. The Use of Hirt's Truncation Error Analysis

The words "finite difference approximations" describe the non-linear algebraic equations which are solved on the computer. These equations approximate the partial differential equations which describe the fluid flows. The deviations are due to the casting of a finite sized mesh over the continuum of interest and the replacement of equations valid at a point with equations applied over discrete intervals. The approach of the discretized solution to the true point solution as the interval approaches zero is the problem of convergence. Concern about this problem often overrides another consideration: the solution obtained is further removed from the desired solution by truncating the infinite Taylor series which are used to construct the finite differences. The truncations of the series are necessary for practical application, but they introduce errors of accuracy which may stabilize or destabilize the solution. Hirt's stability analysis quantifies the truncation errors.

An examination of equation (5-32) illustrates the result of the truncation analysis. An additional term $-U^2 \frac{\Delta t}{2} \frac{\partial^2 W}{\partial Z^2}$ plus higher order differentials not shown are present in addition to the original partial differential equation. Furthermore, the error is always negative in sign, hence destabilizing. In fact, although attempting a numerical solution with $\Delta t \leq 2\nu/U^2$ as specified by (5-33b) may give a stable solution, the second order error is present for any finite Δt.

Harlow and Amsden (25) suggest that the error may be removed by including diffusion terms identical in magnitude to the errors but of opposite sign in the finite difference equations. For example, based

on the result (5-32), the coefficient ν in the finite difference equation (5-2) should be replaced by $\left(\nu + U^2 \frac{\Delta t}{2}\right)$, and the error is automatically removed. Note that the error contains U^2 which was held constant for this example. If U^2 varied through time and space, the correction must be calculated at each time level and uniquely at each grid point. But (25) first introduces these as partial derivatives in the original differential equations. It seems more appropriate to start with the original equations and introduce the corrections after the errors have formed in series truncation. This also avoids the insertion of corrections on the corrections.

The application of Hirt's truncation error analysis to the continuum equations is illustrated by an example of the one dimensional unsteady equations for mass and mementum.

$$\frac{\partial \rho}{\partial t} = - \frac{\partial}{\partial Z} (\rho U) \qquad \frac{\partial (\rho U)}{\partial t} = - \frac{\partial}{\partial Z} (\rho U^2 + P + q)$$

where q is an artificial dissipation as described in equation (2-3). Only the stability of the finite differenced mass equation is examined. This is forward differenced in time and uses a centered space difference first explicitly, then implicitly.

The explicit form is

$$\frac{\rho_j^{n+1} - \rho_j^n}{\Delta t} = - \left[\frac{(\rho U)_{j+1}^n - (\rho U)_{j-1}^n}{2 \Delta Z} \right] \tag{5-36}$$

The left side is in a rearranged Taylor series

$$\frac{\rho_j^{n+1} - \rho_j^n}{\Delta t} = \frac{\partial \rho}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 \rho}{\partial t^2} + \frac{(\Delta t^2)}{6} \frac{\partial^3 \rho}{\partial t^3} \cdots \tag{5-37}$$

The right side is

$$-\left[\frac{(\rho U)^n_{j+1} - (\rho U)^n_{j-1}}{2\Delta Z}\right] = -\left[\frac{\partial(\rho U)}{\partial Z} + \frac{(\Delta Z)^2}{6}\frac{\partial^3(\rho U)}{\partial Z^3} + \cdots\right] \qquad (5\text{-}38)$$

All the above differentials are evaluated at n and j. Substituting these two equations into (5-34) and discarding terms higher than $O[\Delta t, (\Delta Z)^2]$,

$$\frac{\partial \rho}{\partial t} + \frac{\Delta t}{2}\frac{\partial^2 \rho}{\partial t^2} = -\frac{\partial(\rho U)}{\partial Z} - \frac{(\Delta Z)^2}{6}\frac{\partial^3(\rho U)}{\partial Z^3} \qquad (5\text{-}39)$$

The second derivative with respect to time is found by substituting the mass and momentum equations

$$\frac{\partial^2 \rho}{\partial t^2} = \frac{\partial}{\partial t}\left(-\frac{\partial(\rho U)}{\partial Z}\right) = -\frac{\partial}{\partial Z}\left[\frac{\partial}{\partial t}(\rho U)\right] \qquad (5\text{-}40)$$

Then the momentum equation is inserted in the brackets after using

$$\frac{\partial P}{\partial Z} = \frac{\partial P}{\partial \rho}\frac{\partial \rho}{\partial Z} = A\frac{\partial \rho}{\partial Z} \qquad (5\text{-}41)$$

where A is the square of the isothermal speed of sound. Performing the spatial differentiation and retaining only the terms with the diffusion form, the result is

$$\frac{\partial^2 \rho}{\partial t^2} \approx (U^2 + A)\frac{\partial^2 \rho}{\partial Z^2} \qquad (5\text{-}42)$$

Likewise

$$\frac{\partial^3(\rho U)}{\partial Z^3} \approx 3\frac{\partial U}{\partial Z}\frac{\partial^2 \rho}{\partial Z^2} \qquad (5\text{-}43)$$

The final result is

$$\frac{\partial \rho}{\partial t} = -\frac{\partial(\rho U)}{\partial Z} + \left[-\frac{\Delta t}{2}(U^2 + A) - \frac{(\Delta Z)^2}{2}\frac{\partial U}{\partial Z}\right]\frac{\partial^2 \rho}{\partial Z^2} \qquad (5\text{-}44)$$

An examination of the diffusive error shows that the first term is always negative, its magnitude increasing with Δt. The second term depends on the sign of $\partial U/\partial Z$, and a flow accelerating down a tube is destabilizing. The general conclusion is that the explicit finite differencing should destabilize as Δt increases, and this agrees with experience.

The implicit difference form of the mass equation is

$$\frac{\rho_j^{n+1} - \rho_j^n}{\Delta t} = - \left[\frac{(\rho U)_{j+1}^{n+1} - (\rho U)_{j-1}^{n+1}}{2\Delta Z} \right] \tag{5-45}$$

The right difference is a double expansion

$$\frac{(\rho U)_{j+1}^{n+1} - (\rho U)_{j-1}^{n+1}}{2\Delta Z} = \frac{\partial(\rho U)}{\partial Z} + \frac{(\Delta Z)^2}{6} \frac{\partial^3(\rho U)}{\partial Z^3} + \Delta t \frac{\partial^2(\rho U)}{\partial t \partial Z} + \cdots \tag{5-46}$$

and

$$\frac{\partial^2(\rho U)}{\partial t \partial Z} = - \frac{\partial^2 \rho}{\partial t^2} \approx - (U^2 + A) \frac{\partial^2 \rho}{\partial Z^2} \tag{5-47}$$

Except for this term, the rest of the analysis is identical to that for the explicit formulation. The result is

$$\frac{\partial \rho}{\partial t} = - \frac{\partial(\rho U)}{\partial Z} + \left[\frac{\Delta t}{2} (U^2 + A) - \frac{(\Delta Z)^2}{2} \frac{\partial U}{\partial Z} \right] \frac{\partial^2 \rho}{\rho Z^2} \tag{5-48}$$

The implicit and explicit truncation errors are identical except for the sign on the first term. However, this term is now always positive and thus stable for all values of Δt. The inherent stability of implicit methods is well known. Although the truncation errors are stabilizers, it would be advantageous to remove them (provided stability is maintained) because they are errors.

The previous example shows the power of Hirt's method in identifying diffusive truncation errors. Since the errors contain velocities, densities, and their spatial gradients, the physical problem may be examined for regions of flow that might cause instabilities. Hirt examined higher order space differentials to predict unstable regions in shock propagation problems. The predictions were verified numerically. An analysis of MAC stability was also successfully applied to a problem describing flow through a sluice gate in a dam. Thus Hirt's theory is substantiated in numerical tests.

D. The Two Dimensional ICE Truncation Errors

The concepts discussed above were applied to the ICE mass and momentum equations, and also to the species equations. The energy equation was omitted because, with mass and momentum established, there was no indication of stability problems in energy until the last few runs of the study.

Viscous and diffusion terms are of higher order and are not analyzed. Nonetheless, the algebra is mountainous. The results for the truncation error analysis are listed below. The original partial differential equation (OPDE) is on the left side of the equation.

$$\text{Mass OPDE} - \left\{(2\theta - 1)\left(V^2 + N_{Eu}A\right)\frac{\Delta t}{2} - \frac{(\Delta R)^2}{4}\frac{\partial V}{\partial R}\right\}\frac{1}{R}\frac{\partial}{\partial R}\left(R\frac{\partial \rho}{\partial R}\right)$$

$$+ \left\{(2\theta - 1)\left(U^2 + N_{Eu}A\right)\frac{\Delta t}{2} - \frac{(\Delta Z)^2}{4}\frac{\partial U}{\partial Z}\right\}\frac{\partial^2 \rho}{\partial Z^2} \qquad (5\text{-}49)$$

-2

$$\text{Radial Momentum OPDE} = \left\{\frac{\Delta t}{2N_{S\ell}}\left[(2\varphi - 1)N_{Eu}\rho A - 3\rho V^2\right] - \frac{(\Delta R)^2}{8}\left[3\frac{\rho V}{R} + V\frac{\partial \rho}{\partial R}\right]\right\}\frac{\partial^2 V}{\partial R^2} = \left\{\frac{\Delta t}{2N_{S\ell}}\rho U^2 + \frac{(\Delta Z)^2}{4}\left[U\frac{\partial \rho}{\partial Z} + \rho\frac{\partial U}{\partial Z}\right]\right\}\frac{\partial^2 V}{\partial Z^2} \tag{5-50}$$

$$\text{Axial Momentum OPDE} = \left\{\frac{\Delta t}{2N_{S\ell}}\left[(2\varphi - 1)N_{Eu}\rho A - 3\rho U^2\right] - \frac{(\Delta Z)^2}{8}U\frac{\partial \rho}{\partial Z}\right\}\frac{\partial^2 U}{\partial Z^2} - \left\{\frac{\Delta t}{2N_{S\ell}}\rho V^2 + \frac{(\Delta R)^2}{4}\left[V\frac{\partial \rho}{\partial R} + \rho\frac{\partial V}{\partial R} + \frac{\rho V}{R}\right]\right\}\frac{\partial^2 U}{\partial Z^2} \tag{5-51}$$

$$\text{Species Mass OPDE} + \left\{\frac{(\Delta R)^2}{4}\left[\frac{\partial V}{\partial R} + \frac{V}{R}\right]\right\}\frac{\partial^2(\rho_K)}{\partial R^2} + \left\{\frac{(\Delta Z)^2}{4}\frac{\partial U}{\partial Z}\right\}\frac{\partial^2(\rho_K)}{\partial Z^2} \tag{5-52}$$

Equations (5-50) and (5-51) omit the stabilizing correction which results from outer iteration on I^{n+1}. The treatment of corrections for time derivatives in the species equation is uncertain and numerical experimentation usually gave instability. Thus equation (5-52) includes only spatial truncation errors.

Note that a term in the mass equation has a coefficient $(2\theta - 1)$ which determines the sign of the term. It is stabilizing for $\theta > .5$ (implicit), destabilizing for $\theta > .5$ (explicit), and it vanishes at $\theta = .5$ (time-centered). The coefficient φ performs a similar function in the momentum error equations.

There is no contradiction that when θ and $\varphi = 1$ and $A \to \infty$ for incompressible flows, an infinite diffusion error takes place. Instead this predicts the infinite propagation of pulses throughout the fluid, which is a basic assumption of the incompressible equations.

The truncation errors in mass and momentum equations are removed to a specified degree by adding a corrector β as mentioned previously. The corrector β is the finite difference analog of the negative of all truncation errors except those with coefficients that vanish with time-centering. The β corrections follow in finite difference form and include the spatial second derivatives to be consistent with the equations of Chapter 4. The cell aspect ratio removes ΔZ, and coefficients are extracted where possible

$$(\beta_M)^n_{i,j} = \frac{1}{2\Delta R}\left[\left\{V^n_{i+1/2,j} - V^n_{i-1/2,j}\right\}\left[\left(\frac{i-1}{2i-3}\right)\rho^n_{i+1,j} + \left(\frac{i-2}{2i-3}\right)\rho^n_{i-1,j} - \rho^n_{i,j}\right]\right.$$
$$\left. + \frac{1}{2}R_A\left\{U^n_{i,j+1/2} - U^n_{i,j-1/2}\right\}\left[\rho^n_{i,j+1} + \rho^n_{i,j-1} - 2\rho^n_{i,j}\right]\right] \qquad (5\text{-}53)$$

$$(\beta_{VR})^n_{i-1/2,j} = \frac{1}{\Delta R}\left[\left\{\left[\frac{3\Delta t}{4\Delta RN_{S\ell}}\right]\left(\rho^n_{i,j} + \rho^n_{i-1,j}\right)V^n_{i-1/2,j} + \frac{3}{16(i-2)}\left(\rho^n_{i,j} + \rho^n_{i-1,j}\right)\right.\right.$$
$$\left.\left. + \frac{1}{8}\left(\rho^n_{i,j} - \rho^n_{i-1,j}\right)\right\}V^n_{i-1/2,j}\left[V^n_{i+1/2,j} + V^n_{i-3/2,j} - 2V^n_{i-1/2,j}\right]\right.$$
$$\left. + \left\{\left[\frac{\Delta tR_A}{64\Delta RN_{S\ell}}\right]\left(\rho^n_{i,j} + \rho^n_{i-1,j}\right)\left(U^n_{i,j+1/2} + U^n_{i,j-1/2} + U^n_{i-1,j+1/2} + U^n_{i-1,j-1/2}\right)^2\right.\right.$$

(this equation (5-54) continued on next page)

$$+\frac{1}{64}\left(U^n_{i,j+1/2}+U^n_{i,j-1/2}+U_{i-1,j+1/2}+U^n_{i-1,j-1/2}\right)\left(\rho^n_{i,j+1}+\rho^n_{i-1,j+1}-\rho^n_{i,j-1}-\rho^n_{i-1,j-1}\right)$$

$$+\frac{1}{8}\left(\rho^n_{i,j}+\rho^n_{i-1,j}\right)\left(U^n_{i,j+1/2}+U^n_{i-1,j+1/2}-U^n_{i,j-1/2}-U^n_{i-1,j-1/2}\right)\Bigg\}R_A\Big[V^n_{i-1/2,j+1}$$

$$+V^n_{i-1/2,j-1}-2V^n_{i-1/2,j}\Big]\Bigg] \qquad (5\text{-}54)$$

$$(\beta_{VZ})^n_{i,j+1/2}=\frac{1}{\Delta R}\Bigg[\Bigg\{\left[\frac{3\Delta tR_A}{4\Delta RN_{S\ell}}\right]\left(\rho^n_{i,j}+\rho^n_{i,j-1}\right)U^n_{i,j-1/2}$$

$$+\frac{1}{8}\left(\rho^n_{i,j}-\rho^n_{i,j-1}\right)\Bigg\}R_AU^n_{i,j-1/2}\left[U^n_{i,j+1/2}+U^n_{i,j-3/2}-2U^n_{i,j-1/2}\right]$$

$$+\left\{\left[\frac{\Delta t}{64\Delta RN_{S\ell}}\right]\left(\rho^n_{i,j}+\rho^n_{i,j-1}\right)\left(V^n_{i+1/2,j}+V^n_{i-1/2,j}+V^n_{i+1/2,j-1}+V^n_{i-1/2,j-1}\right)^2\right.$$

$$+\frac{1}{64}\left(V^n_{i+1/2,j}+V^n_{i-1/2,j}+V^n_{i+1/2,j-1}+V^n_{i-1/2,j-1}\right)\left(\rho^n_{i+1,j}+\rho^n_{i+1,j-1}-\rho^n_{i-1,j}-\rho^n_{i-1,j-1}\right)$$

$$+\frac{1}{16}\left(\rho^n_{i,j}+\rho^n_{i,j-1}\right)\left(V^n_{i+1/2,j}+V^n_{i+1/2,j-1}-V^n_{i-1/2,j}-V^n_{i-1/2,j-1}\right)$$

(this equation (5-55) continued on next page)

$$+ \frac{1}{16(2i-3)} \left(\rho^n_{i,j} + \rho^n_{i,j-1}\right) \left(V^n_{i+1/2,j} + V^n_{i+1/2,j-1} + V^n_{i-1/2,j} + V^n_{i-1/2,j-1}\right)\Bigg\}$$

$$\left[U^n_{i+1,j-1/2} + U^n_{i-1,j-1/2} - 2U^n_{i,j-1/2}\right]\Bigg\} \tag{5-55}$$

$$(\beta_{MK})^n_{i,j} = \frac{1}{4\Delta R}\Bigg[\left\{\left[V^n_{i+1/2,j} - V^n_{i-1/2,j} + \frac{1}{(2i-3)}\left(V^n_{i+1/2,j} + V^n_{i-1/2,j}\right)\right]\right\}\left[(\rho_K)^n_{i+1,j}$$

$$+ (\rho_K)^n_{i-1,j} - 2(\rho_K)^n_{i,j}\right] + \left\{\left[R_A\left(U^n_{i,j+1/2} - U^n_{i,j-1/2}\right)\right]\right\}\left[(\rho_K)^n_{i,j+1}\ (\rho_K)^n_{i,j-1} - 2(\rho_K)^n_{i,j}\right] \tag{5-56}$$

Since the error analysis is by nature approximate and the correctors must themselves be finite differenced, the use of the full correctors can potentially remove enough diffusion to destabilize the calculation. To avoid this problem, the collections of terms in the brackets { } have coefficients read as input into the computer program. The coefficients can be used in the normal sense, zero to full correction requiring coefficients in the range zero to one. But a number greater than 1.0 can be used. Then the amount greater than 1.0 is applied to the absolute value of the terms in the respective brackets { }. This provides a small amount of positive diffusion to move the problem away from a region of marginal stability.

As indicated by the indices, the corrections are explicit and vary spatially. The computing time and storage for the corrections are good investments if the method stabilizes the computations. As will be seen, this is often the case.

Besides the β correctors, stability is assured by setting an upper limit on Δt of

$$\Delta t < \frac{1}{4} \frac{\rho}{\mu} \frac{(\Delta R)^2 (\Delta Z)^2}{(\Delta R)^2 + (\Delta Z)^2} \tag{5-57}$$

This is a two dimensional analog to equation (5-5). Smaller values of Δt were always used in practice, because numerical trials indicated that they were needed to provide good resolution of the time dependent aspects of the solutions.

The results of the numerical tests will be presented and discussed next.

Chapter 6

NUMERICAL RESULTS AND DISCUSSION

A. Introductory Comments

This chapter presents the results of a sequence of numerical tests of the basic ICE algorithm and its modification and extension. This includes the addition of outer iteration using all first order terms in the hybrid function, the addition of the β stability correctors, and the inclusion of multispecie diffusion and chemical reaction. The numerical solutions are compared with analytical and other numerical solutions where possible. New transient and steady state solutions are shown.

The sequence of problems solved proceeds in the direction of increasing complexity. Initial tests were performed on three transient problems that were essentially one space dimension in nature. Thus appearance of significant quantities in the second spatial dimension would indicate errors in the computer code or a failure of the algorithm. Both incompressible and fully compressible problems were run.

The first two-dimensional problem was one describing the boundary layer buildup in the entrance region of a tube after flow is abruptly started. Attempts at calculations of transient coaxial flows led to major instabilities. This prompted inclusion of the β truncation

error corrections with dramatic improvement. Numerical experiments showed the advantages of the ADI method in inverting the $\tilde{P}$ matrice, and this was adopted.

The addition of multicomponent capability caused new stability problems which were treated by developing an implicit solution of the species equations. Finally chemical reaction was included. Solutions for dilute component reaction were run without difficulty. Later computations of reacting flows with strong coupling between chemical reaction and the fluid dynamics showed serious instability which could not be treated within the scope of this effort.

The boundary conditions for each problem are discussed briefly as a supplement to the detailed presentation in Chapter 4. Values of time step size, grid size, dimensionless groups, and so forth are found in Table 1.

As noted in Chapter 4, the solution accuracy was measured by the error in the mass conservation equation over one time increment. This in turn is controlled by the convergence of the $\tilde{P}$ iteration. The convergence is tested by

$$2\,\frac{\left|\tilde{P}^{Q+1} - \tilde{P}^{Q}\right|}{\left|\tilde{P}^{Q+1}\right| + \left|\tilde{P}^{Q}\right|} < \varepsilon \qquad (6\text{-}1)$$

The exponent on ε specifies the decimal place on $\tilde{P}$ which must converge, regardless of the exponent on $\tilde{P}$. Thus if $\varepsilon = 10^{-3}$, $\tilde{P}$ must converge to three decimal places, such as 2.345×10^{-8}. Generally ε used varied from 10^{-4} to 10^{-6}.

B. Computer Facilities and Computer-Drawn Plots

The solutions were carried out on the NASA Lewis Research Center IBM 360/67 computer. This is a time sharing system (TSS), using virtual memory and a paging system to locate and move data in and out of core storage. The Lewis computer is duplexed with two central processing units (CPU) and a combined core storage of 2.5 million bytes (4 bytes per word). Virtual storage essentially provides unlimited storage during execution.

The computations were run in double precision, giving approximately 13 decimal places of significance. The time sharing mode was employed to permit periodic interruption and examination of the progress of the solution. During the interrupts, values of any variable could be displayed on the terminal, and certain operating parameters such as the time step size could be changed. Due to the time sharing mode, CPU run times cited herein are approximate.

Output was written into datasets during the run which were dumped onto the IBM 1403 printer and/or onto a 9 track tape after the run was done. The tape was later read into a plotting program which used a set of library subprograms to generate plots on a CDC Model 280 cathode display unit. The video plots were photographed by a 35 mm single frame camera, and plots were made directly from the film. The basic library subprograms are described by Kannenberg in reference (33).

The dependent variable data taped for plotting were triply subscripted to denote R, Z, and t values. Plots were made in the time sharing mode by first selecting a value for one of the three independent variables. The plotting program then arranged the appropriate data

to form a two dimensional array and fitted the array to a surface using a procedure by Akima (1). This surface of dependent variable values could be projected as a three dimensional plot using Canright and Swigert's subprograms (10). The surface could be sliced horizontally to produce contour curves plotted by a routine by Lawson, Block, and Garrett (36). Slices through the other two directions produced X-Y type plots.

A pictorial plot could be made to display the total velocity field at a fixed time. The magnitude and direction of the flow is shown by the size and direction of arrows distributed over the R-Z field. As an alternative, only direction is shown to clarify the flow direction where the magnitude is small. The bases of the arrows locate the position of the velocity being represented.

The plotting program was developed parallel with the numerical experimentation. Thus the figures which describe the early computations are hand drawn, and the computer plots are used for the coaxial entry problems. It is emphasized that the Akima surface fit gives a very excellent fit which essentially passes through all the input points. Thus the results are not smoothed by this fitting procedure, but rather made continuous.

C. Startup of Incompressible Laminar Flow in an Infinite Tube

This problem is characterized by axial velocities which are a function of R and t only. No radial velocities occur. A constant upstream pressure is suddenly imposed on a tube containing an incompressible fluid at rest. A linear axial pressure gradient results and is

maintained. The fluid starts motion as slug flow, but the no slip wall condition causes transition to parabolic Poisieulle flow at large times. The analytical solution can be found in any standard fluid dynamics or transport text such as Bird, Stewart, and Lightfoot (4).

This problem was selected as a starting point because (1) the simplicity of the problem allowed easy tracing of program errors, and (2) the numerical results could be compared with the analytical solution and also with an available MAC numerical solution. To make the last comparison, the fluid was specified as blood at 310° K, $\rho = 1.05$ gm/cm^3, $\mu = .04$ gm/(cm-sec). The Reynolds number was 205. Boundary conditions are straightforward, with $Q = 0$ due to incompressibility giving the necessary velocities outside the input, output, and wall boundaries.

The results are shown in figure 4 as radial profiles of the axial velocity at select times. The solid line is the analytical solution and the ICE solution is shown as discrete points. Three cell mesh sizes were run to check the convergence behavior, which should be sensitive only to the radial solution. The cell aspect ratio was adjusted to keep the tube length constant. The numerical and analytical solutions compare very well in time and space. Remarkably, even the 4 x 4 grid gives a reasonable solution, although numerical values are somewhat higher at later times. The coarse grid has difficulty representing the velocity gradient near the wall. On the first time step, only the upstream input pressure was specified, and the $\tilde{P}$ iteration produced the correct linear axial gradient. The $\tilde{P}$ convergence criterion was set to the rather severe value of 10^{-8}, resulting in

negligible radial velocities and continuity errors. CPU times ranged from three to eight minutes. This test of ICE on a simple incompressible flow was quite successful.

D. The Shock Tube Problem

A fully compressible flow occurs when a diaphram separating high and low pressure compartments in a tube is abruptly removed. This condition exists when the diaphram of a shock tube is cleanly burst. The gas is air at 20° C. A shock wave propagates downstream through the low pressure gas and a rarifaction propagates upstream. The contact surface, a density discontinuity which denotes the original position of the diaphram relative to the gas, moves downstream with the gas. Initially, the temperature is uniform. It rises at the shock and falls at the rarifaction.

The fluid dynamics of a shock tube can usually be well represented as an axial transient flow. Only the formation and propagation of the waves to the tube ends was computed. The problem was formulated with no slip walls and reflective ends. A run with zero viscosity gave identical results over the short time period considered (1/2 millisecond). Inviscid theory can be used to predict the positions of the various waves and the levels that the variables should approach upon passage of the waves.

The results of the primary variables, P, ρ, U, and I, are shown in figures 5a through 5d. CPU time for this run was 3 minutes. The pressure rarifaction should not be a discontinuous wave, but even so, both the rarifaction and the shock show considerable numerical smearing. The shock shows overshoot and the pressure trailing the

rarifaction indicates some numerical oscillations. No β corrections had yet been put into the numerical program. The oscillations are prominent in the density profile and especially in the internal energy profile. The energy oscillations are due in part to the computation being run with no thermal conduction. It was not realized at that time that thermal conductivity damped such oscillations as effectively as viscosity [see Harlow and Amsden (26)]. In addition, Hirt (29) predicted that numerical instability could arise in this region due to truncation errors associated with third and fourth order differentials. Such differentials are important due to the steep gradients in the shock tube problem.

Despite these difficulties, the waves at any position damp as the fronts moved away. The profiles of all variables show the proper qualitative and quantitative behavior. The density profile has four levels due to the contact surface, and the internal energy rises in the compression zone and falls in the rarifaction zone. The positions and levels associated with the fronts as calculated by the ICE method compare very well with inviscid theory, even if oscillations are present. As discussed in Chapter 2, shock calculations are usually made with an added artificial dissipation. Such a term is not used in the present version of ICE, and some instability is to be expected. Thus the applicability of the ICE technique to compressible flows seems well demonstrated.

Additional numerical experimentation was performed. Figure 6 shows the effect of using a smaller time step size. The solid line is the same as figure 5a. The smaller Δt provides steeper fronts and better

stability. A small Δt offers greater resolution of temporal behavior, hence the steeper fronts. This is a valuable characteristic of the ICE method. The time step size is chosen to resolve phenomena of interest. Thus a small Δt is used to examine shocks, and a large Δt is used for very slow flows where acoustic behavior is unwanted. A smaller time step also provides a better computation for an impulsive start as in the shock tube problem. This affords greater stability later in the course of the computation, as shown in figure 6.

The expanded hybrid function which retained the other first order term I^{n+1} was also tested. The results are shown in figure 7. The solid curve is the same as figure 5a since no iterations on I^{n+1} means I^{n} is used, and the extra term vanishes. No discernible difference of the pressure profile is detected after one replacement of I^{n+1}. Remarkable stability improvement is seen, but the added numerical diffusion smears the wave fronts, which is undesirable. For this reason and because even one outer iteration on I^{n+1} almost doubles the CPU time, the expanded hybrid function was not used for the remainder of this study. Its use is recommended only as an aid in suppressing numerical instability.

The effect of time centering the equations by setting $\theta = \varphi = .5$ is shown in figure 8. Both implicit and time centered pressure profiles are shown for two times after the diaphragm is removed. As predicted by Hirt's stability analysis, time centering steepens the fronts at the expense of stability. This is especially true for the region just behind the shock. However, the waves do die out at fixed Z positions.

Longer calculations would be needed to verify whether or not the numerical oscillations directly behind the shock front will grow unbounded.

As a final experiment, a one-dimensional β truncation error correction was inserted in the mass equation to see if the oscillation amplitude would decrease. Figure 9 displays the results of a time centered test. The exact correction did not affect the oscillations, but adding four times the absolute value of the correction cut the perturbations in half. Such a massive correction is unjustifiable theoretically, but it did show the nature of the β corrector. Note that only the unstable region of the profile was affected. The shock and rarifaction fronts are the same, as are the levels behind them. To properly stabilize the shock tube calculation, mass, momentum, and energy correctors are necessary, including the important higher order differentials.

E. Incompressible Tube Flow with an Oscillatory Upstream Pressure Perturbation

This problem was used to test the ICE program with oscillatory flow. A tube holding an incompressible fluid is subjected to an upstream axial pressure which has mean and oscillatory components. The oscillatory component has an amplitude which is 10% of the mean component so no net flow reversal occurs. However, if the mean flow is subtracted from the total flow, the resulting velocity perturbations do reverse direction.

An analytical solution for the steady cyclical state of this problem has been carried out by Uchida (50). The axial velocity is assumed independent of Z, making linearization possible. The analysis shows that as the frequency of the pressure oscillation increases, the velocity oscillations act more like oscillating plug flows.

The boundary conditions for the numerical solution of this problem are the same as for startup of incompressible flow in an infinite tube. The exception is the oscillatory upstream pressure. The numerical solution was run first by assuming that the flow was already at the steady cyclical state. The initial condition on axial velocity was taken from Uchida, and the computation was carried out for a few cycles to allow comparison with Uchida's solution.

The results are shown in figures 10 and 11 for dimensionless frequencies of 3 and 10 and a fractional pressure perturbation of 0.1. The fluid was air at 45° C. The ordinate is the velocity perturbation amplitude that is superimposed on the mean flow, normalized by the fractional pressure perturbation impressed on the flow. A comparison of figures 10 and 11 indicates that the lower frequency produces an oscillatory flow which is strongly affected by shear stresses, whereas the higher frequency causes a flat profiled, plug-type flow that is dominated by inertia effects. In both figures the numerical and analytical solutions compare well, but for a dimensionless frequency of 10, the numerical solutions near the wall lag slightly.

There are some errors in producing the plots. The ICE program calculates the total velocity including the mean and perturbation.

The steady state Poisieulle solution, calculated from another computer program, is then subtracted from the total to give the velocity perturbation. Normalization by the fractional pressure perturbation causes a multiplication by 10. Thus an ordinate value of .01 on figure 11 actually represents only a 0.1% variation of the total flow. Therefore a small difference generated in processing the numbers gives a sizeable deflection on figure 11, and in fact the analytical and numerical solutions compare very well.

Further experiments on convergence showed that a reduction of grid size from 20 x 5 to 10 x 5 gave a poorer comparison between analytical and numerical solutions for the lower frequency. Evidently a ΔR near .05 is necessary to represent boundary layer effects. Reducing the time step size from $\pi/30$ to $\pi/60$ improved the higher frequency solution, but a further reduction in Δt offered little gain.

An extension to the oscillating flow problem was made by starting with the flow at rest. The upstream pressure was suddenly raised to the higher oscillating value, coupling the normal laminar startup with the oscillatory flow startup. A run starting from rest and continuing until the steady cyclic state was reached took 15 minutes of CPU time. The velocity perturbations were found by subtracting the transient mean flow predicted by an analytical solution.

Figure 12 shows a velocity perturbation profile at the same cycle time, $2n\pi t$, for a reduced frequency of 5. The oscillatory component almost reaches the cyclic solution within one cycle. The steady cyclical numerical values are slightly high. As discussed before, the

ordinate scale is expanded, and the constant difference between the numerical and analytical values along the tube radius suggests a small error in data reduction.

Figure 13 shows the final cyclic solution which proceeded from a flow starting from rest. A number of cycle times are represented. The comparison with the analytic solution is good. All numerical values again are slightly high, indicating a data processing error. In conclusion, the ICE method handles oscillatory flows quite well provided that the grid cell size is small enough ($\Delta R = \sim.05$) and a suitably small time step is chosen to give adequate transient resolution.

F. Transient Incompressible Entry Tube Flow

The entry problem is dependent on two space dimensions. A uniform velocity is abruptly applied to the entrance of a tube filled with an incompressible fluid initially at rest. The fluid behaves like plug flow at first, but a boundary layer begins to build up. The boundary layer trails from the entrance edge of the tube. At steady state the uniform input flow changes to Poisieulle parabolic flow for a sufficiently far downstream position. The steady state solution to this problem has been treated by Hornbeck (32), and Friedman, Gillis, and Liron (19), among others.

The numerical version of this problem is considerably different from the previous test problems in another respect. Previously, the pressure condition was specified and this provided the driving force. In this and the remaining problems, the input flow is specified. This flow drives the fluid through the tube and the proper pressure field is

computed to accelerate and turn the internal flows. The momentum equation is applied at the input boundary to compute the pressure gradient across the input. Thus the mean value at the input can "float" during the $\tilde{P}$ iteration until the entire pressure field has the proper curvature.

To test the ability of the $\tilde{P}$ iteration to give the correct pressure field, an artificial problem was run as follows: the incompressible fluid is initially moving through the tube in axial plug flow. At $t = 0$, the entrance and exit are abruptly blocked. By the end of one Δt, the fluid should be essentially stopped. The test was run with free slip walls. In one $\Delta t = 10^{-4}$, the flow was reduced from 1.000 to less than 10^{-3}. When the test was rerun with a no-slip wall, the bulk flow was again essentially stopped. But the asymmetric boundaries (no-slip at the wall and free slip at the centerline) caused a very weak eddy circulating down the axis and up the wall. Although trivial, this result is not physically inconsistent. The test of the $\tilde{P}$ iteration was considered a success.

The results of the entry flow problem are shown in figures 14 through 18 for air at 45° C. Figure 14 shows the steady state axial distribution of the axial velocity at several radii. In Hornbeck's paper, velocity values are given for different radii, but linear interpolation can be used to give check points at $R = .05$ and .95. Agreement is good. The interpolation of the points at the lowest Z position is the least accurate, and those points show the most disagreement. Since a horizontal profile means that the flow is constant

with length, the figure shows the growth of the steady boundary layer toward the tube centerline as the flow progresses downstream.

Figure 15 is a plot of the steady-state radial profiles of axial velocity at various Z positions. The transition from the flat input velocity profile to the parabolic output profile is clearly seen. Much of the transition takes place near the input. Friedmann et al. (19) have observed that near the entrance, the profiles show a maximum which does not occur on the centerline. This effect is seen on the $Z = .667$ profile. Friedmann et al. attribute it to the fact that the leading edge of the tube propagates pressure signals upstream so the flow begins to turn before coming to the tube. This is more predominant at the low Reynolds numbers. An input boundary condition cannot describe these complex upstream effects. If the edge wall boundary condition is set so that the constant input velocity extends to $R = 1.0$ instead of $R = .95$, exactly the same solution results.

The radial flows at steady-state are shown in figure 16. A pressure spike develops at the leading edge of the tube to turn the flow toward the tube axis. The radial velocities decay rapidly with length and vanish where the flow is fully parabolic. At $Z = .33$, the radial velocity has a maximum magnitude which is 13% of the input axial velocity.

Transient axial velocity profiles are shown in figures 17 and 18. Steady state is achieved quite rapidly near the input boundary and near the wall. If a plot like figure 18 is constructed at $Z = 20$, the resulting profiles are almost identical to those of figure 15.

The ICE method works quite well for the entry problem. It can provide additional information on radial flows and transient flows that simpler analyses cannot provide. CPU time was 26 minutes.

G. Stabilization of Coaxial Flow Calculations by the β Truncation Error Corrections

One of the major goals of this study was to incorporate multi-specie flows. Successful computation of coaxial entry flows was an essential step to that end and a worthwhile goal in itself. The coax injector has been a standard mixing device for some time. A number of studies of steady-state flows such as Ghia et al. (22) and Weinstein and Todd (54) have supported conceptual designs of gaseous nuclear rockets. Flow recirculation is possible if the ratio of the center tube and annulus velocities becomes very large or very small. This has been experimentally studied by Warpinski, Nagib, and Lavan (53). To this writer's knowledge, no numerical studies of transient or short tube coaxial flows have been published.

Besides varying the input flows, two extremes of tube length suggest themselves. An extremely short tube allows resolution of the flows very near the entry point. This can not be done for long tubes due to the limitation on the number of mesh cells which will give a reasonable computation time. The other extreme is a long tube of sufficient length such that the output flow is parabolic.

Wake effects from the boundary separating the two entry flows have been studied by Paris (41) for parallel plates. Seider (47) was unable to use a mesh aspect ratio greater than .05 without incurring

instability in this computation of tube flows with coaxial parabolic entry. He attributed this to wake problems.

A short tube problem was attempted for air at 45° C, with an input velocity ratio of 2.0. The fluid is initially at rest and is subjected to a step coaxial input at $t = 0$. The center and annular flows were each uniform. The tube length was equal to its radius. A 20 x 20 mesh was used with square cells. This mesh is the largest that could be used with the available computer speed. Figure 19 shows the catastrophic instability which occurred when this problem was attempted. A long tube run was then tried with $R_A = .05$. A smooth steady state solution was obtained, but the transient solution showed numerical oscillations which eventually damped.

At this point the β corrections were added to the computer program and the short tube problem was rerun. The coefficients on β_M and $\beta_{VR} = \beta_{VZ}$ (called β_V) were set equal to 0 and 1.01, respectively. The results were remarkable. All numerical oscillations vanished and the problem was computed to steady state. Figure 20 shows results which may be compared to figure 19 for $t = .07$ and .10. The scales on these computer drawn plots are set by the graphics program and differ. It is seen that the stable portions of figure 19 compare exactly with the similar parts of figure 20. These results were most encouraging.

H. Short Tube with Coaxial Entry - Velocity Ratios = 2 and .05, Velocities = 10 and 20 cm/sec

The next series of coaxial problems (Sections H, I, and J) were run using air at 45° C as the single fluid. The coaxial runs were

identical whether run as compressible or incompressible fluids. At these slow speeds the compressibility effects should be negligible, and they were. As mentioned before, the upstream pressure was computed by applying the momentum equation at the input. The downstream pressure was fixed. Input and output density was computed using the equation of state. The problems were started at the time steps shown in Table 1. The time steps were increased throughout the run using the TSS mode of operation. The progress of the run was monitored to guide the increases on Δt.

The formidable problem of presenting the results of these transient, complex flows was aided by the computer plotting routines. About 50 to 100 plots were made for each run. Many of these were difficult to interpret without extensive cross comparison. It was decided to show the transient results in a semi-quantitative manner using three-dimensional (3-D) plots at select times. The steady state plots are mainly radial profiles at select axial positions.

Figures 21a through d show the axial velocity at several times for the velocity ratio of 2. The axes of perspective are given by the minature X-Y-Z coordinate system on the right. For all 3-D plots the Z axis denotes the magnitude of the dependent variable, the Y axis points in the radial direction, and the X axis points in the axial direction. The maximum and minimum values along each axis are given. The scale of the figure changes from plot to plot. Viewing these figures in the time sequence shows that initially the coaxial inputs produce a uniform output. With the passage of time, the high center

velocity and low annular velocity move toward the exit while transferring momentum at their interface and at the wall.

Figure 22a shows a 3-D pressure plot right after the flow starts, and figure 22b shows the pressure field at steady state. Note that the vertical scales differ, and in examining figure 22a, note that the pressure spike in the center of the input boundary points down. All the 3-D plots have no hidden lines. A strong gradient is set up at the inside wall of the annulus, hereafter called the annular edge. The gradient forces the high velocity center input to turn outward and raise the velocity in the annular region. At the same time the leading edge of the wall is forcing fluid toward the centerline. At steady state the flow field is established, and the fluid already is accelerated. Then a strong driving force is no longer needed and the radial pressure gradient is considerably weaker, as seen in figure 22b.

The radial flows are so complex that it is impossible to find a figure that can present them properly. In brief, the initial flow is mostly outward with the maximum at the annular edge. At steady state, the center tube radial flow is weakly inward and the annular flow is strongly inward, which decreases the radial gradient of axial velocity between the two streams.

The steady state radial profiles of the axial and radial velocities are displayed in figures 23a and b for select axial positions. Figure 23a shows the rapid dissolution of the steep velocity gradient separating the two fluids. This region of high shear stress promotes rapid momentum transfer. The complexity of the radial flows is seen in

figure 23b. The wall is always moving the fluid inward, but there is a small outward flow near the entrance at $R = .5$ which is raising the axial velocity in that region. CPU time was 36 minutes.

To further characterize the entry region for coaxial inputs, the short tube problem was run with the higher velocity entering through the annulus. Since the wall drag tends to force the fluid toward the centerline, the velocity ratio of .05 was expected to cause larger forces to be impressed on the fluid. This could make the numerical solution more difficult. When the problem was run with $\beta_V = 1.0$, it became unstable at $t = .16$. Evidently the β corrector removed some stabilizing errors. Some numerical experiments were run increasing β_V above 1.0, since this adds the absolute values of the error corrections, cancelling negative corrections, and improving stability. Tests were made with β_V as high as 2.0, which changed the third significant figure in velocity values after the computation had proceeded part way through the problem. A value of $\beta_V = 1.5$ caused a maximum change of one digit in the fourth place. This β_V value was used and the problem ran to steady state ($t = .26$) successfully.

The results are shown in figures 24 and 25. Figures 24a through 24d are 3-D plots of the axial velocity. As with the previous problem, the initial response of the fluid to the step impulse input is to move in plug flow. This is seen in figure 24a. Progressing from upstream to downstream, the axial velocity rises in wave along the tube centerline and falls in a wave along the annulus centerline. This condition is the inviscid solution to the initial conditions. As time progresses these waves convect downstream and the flow slants toward the centerline.

Plots of the steady state velocities are shown in figures 25a and b. Although the maximum axial velocity in the annulus is increasing, the mass flow is decreasing. The only outward flow at any time in the problem occurs just downstream of the annular edge, and the pressure profiles show the wall and annular edge effects. About 30% of the way down the tube, the initial turning of the flow has essentially been completed.

I. Long Tube with Coaxial Entry - Velocity Ratios = 2 and .05, Velocities = 10 and 20 cm/sec

The method was applied next to a long tube using cells twenty times longer than wide. This provides less detail in the axial direction and also leads to smaller radial velocities, since the first computed value is already one tube radius downstream. These radial cross flows can cause nonlinear instability, so their lessening was expected to give favorable stability.

Since the tube length is ten diameters, the exit flow should approach a parabolic profile for the low Reynolds numbers of these problems. The length of ten diameters is near the practical maximum for the computer used, due to speed and storage limitations. A lower aspect ratio would be less accurate. If only steady state solutions were desired, the appropriate steady equations could be used with a numerical marching technique and any length could be used. Ghia et al. (22) used this method to follow solutions for over 250 diameters downstream.

The results are shown in figures 26 and 27 for a velocity ratio of 2. Transient runs are shown in the series of figure 26. The initial impulses convect down the tube, but they are overwhelmed by the boundary layer buildup in the long tube and associated flow increase on the axis. The steady state values are shown in figures 27a through c. Full parabolic axial flow is almost achieved at $Z = 20$, and the pressure field becomes almost linear with tube length after $Z = 2$. CPU time was 85 minutes for this run.

No transient 3-D figures are shown for the long tube with an entry velocity ratio of .05. Figures 21, 24, and 26 provide sufficient insight into how the flow develops. The steady state values are shown in figures 28a to c and are consistent with the short tube results. Due to the initially low centerline velocity, the axial velocity at $Z = 20$ is not yet parabolic, and the centerline velocity is only 90% of the final value. CPU time was 34 minutes.

In summary, the ICE method with β stabilization allowed calculation of transient coaxial-entry problem and short tube problems which were not previously available. Since these flows were quite nonlinear, the versatility and stability of the method is well demonstrated.

J. Center Jet Flow into a Larger Tube - Velocity Ratio = ∞, Velocity = 20 cm/sec

A complete parametric survey of velocity ratio, area ratio between the center tube and annulus, and Reynolds number was not run for the coaxial entry problem, nor was such an effort appropriate for this study. However, recirculation zones represent another increase in

complexity of flow and thus are of interest to this study. Recirculation in coaxial jets was found by Warpinski and Nagib (53) to exist above velocity ratios of 8 for Reynolds numbers greater than 400. It was decided to run a coaxial problem with no flow in the annulus. This gives an infinite velocity ratio and is the same as a jet issuing into a larger diameter tube. A tube of intermediate length, five radii long, was chosen. Previous coaxial tests were made with the fluid assumed incompressible, then checked in part with an identical compressible fluid. The calculation described in this section was run to steady state for both compressible and incompressible identical fluids and gave identical results.

The transient solution is shown in figure 29 as pairs of diagrams which show the velocity field and the flow direction at four times. The figures labeled "Velocity Field" have arrows whose magnitude and direction refer to flow values located at their origin. The arrows are normalized to the maximum velocity in the field and that value is given at the base of the figure. Figures labeled "Flow Direction" use arrows whose lengths are independent of the flow velocity to show the direction of small magnitude flows. At $t = .04$ the local impulse of the jet is seen dispersed over the entire tube. But by $t = .1$ a small amplitude recirculation zone was formed in the annular portion of the tube. Figures (e) through (h) show the growth of the eddy down the tube. Figure (h) shows that the jet has filled the tube at steady state, inasmuch as the eddy is contained within the tube with no back flow into the tube along the wall at the downstream boundary. However,

figure 29g shows that most of the fluid flow is still in the center of the tube.

The steady state velocity profiles are shown in figure 30. The negative axial velocities between R = .6 and the wall indicate the recirculation zone. No signs of instability occurred during the run, and the ICE method handled the recirculation flows quite easily. CPU time was about 80 minutes.

K. Coaxial Parabolic Entry Flow with Specie Diffusion and Chemical Reaction

The success of the stabilized ICE technique with all the problems previously described encouraged the incorporation of the species conservation equations into the algorithm. The study by Seider (47) was taken as the comparison case. This author examined the steady flow of fully developed coaxial streams into a larger tube where the fluid was nitrogen with tracer quantities of hydrogen in the center jet and iodine in the annular jet. As the streams mix, hydrogen iodide is formed irreversibly.

The use of trace quantities decouples the species equation from the mass momentum equations since negligible heat is released and inconsequential density changes occur. Seider transformed the primitive flow variables to stream function and vorticity, and rearranged the two species equations so that the mixing without reaction could be calculated first, and the results used in a reaction equation. The latter technique is valid only for two-component reactions.

The sequence used in solving the species equation in the present program was described in Chapter 4 and is applicable to multicomponent reactions. This method evolved from numerical experimentation. Initially it was planned to solve the species equation explicitly over the full time increment, but this scheme was unstable. Then the equation was made entirely implicit and solved repeatedly until the values converged. It provided more stability, but it appeared that the reaction equation was overshooting on each time step. Finally the algorithm was put into its present form with only the convective fluxes implicit and the diffusive terms explicit. The matrix is inverted by the ADI method and the time steps subdivided for more stability.

Seider's problem was run for the condition of total Reynolds number = 496 (N_{Re} = 248). There were some inconsistancies in the input conditions which were treated by setting the pressure quite high to give .005 mole inputs of the tracers. The problem was run as an unsteady case with the parabolic entry suddenly impressed upon the tube whose contents were initially at rest. The initial fluid composition within the tube was arbitrarily set as a core of nitrogen plus hydrogen and an annular cylinder of nitrogen plus iodine. After the problem came to steady-state, the reaction was "turned on" by resetting an index within the time-sharing computer mode, and computation proceeded to the new steady-state. CPU times were difficult to estimate under these circumstances but probably ranged from one to two hours.

The results are shown in figures 31 through 34. It is impossible for a 20 x 20 grid to exactly duplicate Seider's entrance

condition because the axial velocity is not defined at the annular edge. Hence the value of U for $Z = 0$ does not go to zero at the annular edge, and furthermore, the annular edge is located at $R = .55$ rather than .563 which Seider used. Nonetheless, the axial velocity profiles appear almost identical to Seider's graphical results. Figure 31b shows radial velocity profiles, which Seider did not report.

The results of the ICE computation of the mixing of the core jet were compared with Seider's results for no reaction in figure 32. The comparison is excellent, giving confidence to both the total fluid dynamics and the species mixing. The results with reaction were less pleasing. Seider computed a fractional conversion of the hydrogen of .17 at $Z = 20$, while the ICE computation gave about .13. In the absence of comparison with data, the relative accuracy of the two numbers cannot be discerned.

The steady state concentration profiles with reaction for H_2, I_2, and HI are shown in figures 33a through c. Except for $Z = 20$, the curves in 33a and 33b look essentially the same since reaction is slow. Seider did not keep track of the hydrogen iodide reaction product in his calculations. Figure 33c shows the concentration profiles for this specie, and some interesting additional information is contained there. The peaks of the profiles shift to lower radius values with increasing Z up to $Z = 8$ because there are small but appreciable radial flows toward the axis over that length (see figure 31b). The strange profile at $Z = 20$ is due to the longer residence time of the fluid near the wall. These fluid elements have more time to react provided that the

hydrogen has reached the wall, which figure 33b shows it indeed has done between Z = 3 and 8. The effect of boundary layer drag on chemical conversion is also shown in the contour map of the HI concentration, figure 34. The higher values and steeper concentration gradients in the upper right of the figure show the wall effect. The shift of the profile peaks toward the axis may also be seen.

Seider reported that instability resulted when cell aspect ratios greater than .05 were tried. To show the superior stability of the present method, a computation with $R_A = .2$ was run and the results are shown in figures 35 and 36. No stability problems ensued, the results are consistent with those of the longer tube, and more detail near the injection point is available. The test was a complete success.

L. Attempted Solution of Coupled Reaction and Fluid Mechanics

At this point it was apparent that some intricate computations and parametric surveys could be made studying the mixing and reaction of small concentrations of reactants in recirculating coaxial flows. However, it seemed more consistent with the rest of the study to attempt the last step, that of coupling reaction and flow by computing the mixing and oxidation of carbon monoxide with air. This problem was of interest to pollution, and the detailed chemical reaction steps had been studied by Brokaw and Bittker (8). They used a mixture representative of automobile exhaust with the assumption that the reactants were initially well mixed. A global rate could be gotten from these calculations (see Appendix B). A similar mixture was used here and the

composition is listed in Table 1. Although only 10% CO was present, a temperature rise of 200-300° F was expected. This could accelerate the compressible flow such that the constant downstream pressure assumed for earlier coaxial runs and justified in Shapiro (48) might no longer be valid. However, that assumption was retained at the start. The paper of Brokaw (7) was used as a basis for simplifying the computation of mixture transport properties, and these properties were provided by Svehla (49). Thermodynamic data were gotten from Gordon and McBride (23).

It was decided to first attempt a solution with no reaction. The problem was started as impulsive uniform coaxial flows of air in the center tube and the hot gas containing CO in the annulus. The tube initially contained air. The calculation proceeded very well to $t = .065$ when disasterous dynamic instability occurred. Figure 37 shows the last temperature profile obtained. It can be seen that instability is present in the profile near the mixing zone. Reducing the time step did not help. At this point the numerical experimentation was stopped.

It is felt that there were two prime reasons for the instability. First, no truncation error corrections had been derived for the energy equation because of the massive effort it entailed and because there was no indication of dire need. Second, it was suspected that more cells in the radial direction were needed for convergence under the severe temperature and density gradients posed by the problem. This difficulty could not be treated within the capacity of the present

computer. These facts plus time restrictions caused the study to be terminated. However, the superior applicability of the method was demonstrated and it is this author's opinion that the method can also be applied to strongly coupled flows.

Chapter 7

CONCLUSIONS AND RECOMMENDATIONS

1) The Implicit Continuous Eulerian (ICE) method has been shown to be applicable to transient, two-dimensional calculations of both high and low speed compressible flows and to incompressible flows.
2) The inclusion of truncation error corrections based on Hirt's stability theory provided excellent stability behavior without distorting the numerical results.
3) The stabilized ICE method permitted computation of coaxial entry problems which were previously impossible due to wake effects.
4) The method successfully treated complex flows such as unsteady recirculation zones.
5) The addition of multicomponent mixing and chemical reaction was successful for flows without strong coupling between reaction and fluid dynamics.
6) The expansion of the hybrid function to include outer iteration on the internal energy caused greater stability at the expense of excessive numerical diffusion and computer time. The use of the expanded hybrid function is not recommended.
7) The iteration of the pressure field using successive over-relaxation (SOR) and an alternating direction implicit (ADI)

technique gave identical results. The SOR method was faster for meshes with square cells while the ADI method was faster with long rectangular cells.

8) The application of the method to problems characterized by strong coupling between reaction and flow was unsuccessful.

9) It is recommended that experimentation with the ICE method continue with flows strongly coupled with reaction. This includes (a) derivation and testing of a truncation error correction for the energy equation, (b) experimentation with time centering for compressible flows, and (c) use of larger grids.

Table 1 - Description of Computations

Problem Type #	Description	Fluid	$\bar{T}^*$ °R	$\bar{P}^*$ (gm/cm sec^2)	$\bar{\rho}^*$ gm/cm^3	$\bar{M}^*$ gms/gm-mole	$\bar{I}^*$ cm^2/sec^2	$\bar{C}^*$ gm-moles/cm^3
1	Incompressible Startup in an Infinite Tube	Blood	310	1.013×10^6	1.050	38.66	-	-
2	Compressible Flow in a Shock Tube	Air	293	1.013×10^6	1.206×10^{-3}	29.00	2.100×10^9	-
3	Steady Incompressible Cyclical Oscillatory Flow	Air	319	1.013×10^6	1.108×10^{-3}	29.00	-	-
4	Incompressible Transient Oscillatory Flow	Air	319	1.013×10^6	1.108×10^{-3}	29.00	-	-
5	Incompressible Tube Entry Flow	Air	319	1.013×10^6	1.108×10^{-3}	29.00	-	
6	Coaxial Entry in a Short Tube. Length=1 radius, Center Tube Velocity=20 cm/sec, Annulus Velocity=10 cm/sec	Air	319	1.013×10^6	1.108×10^{-3}	29.00	2.286×10^9	-
7	Coaxial Entry in a Short Tube. Length=1 radius, Center Tube Velocity=10 cm/sec, Annulus Velocity=20 cm/sec	Air	319	1.013×10^6	1.108×10^{-3}	29.00	2.286×10^9	-
8	Coaxial Entry in a Long Tube. Length=20 radii, Center Tube Velocity=20 cm/sec, Annulus Velocity=10cm/sec	Air	319	1.013×10^6	1.108×10^{-3}	29.00	2.286×10^9	-
9	Coaxial Entry in a Long Tube. Length=20 radii, Center Tube Velocity=10 cm/sec, Annulus Velocity=20 cm/sec	Air	319	1.013×10^6	1.108×10^{-3}	29.00	2.286×10^9	-

Table 1 - Description of Computations (Continued)

Problem Type #	Description	Fluid	$\overline{T}^*$ °R	$\overline{P}^*$ ($gm/cm\ sec^2$)	$\overline{\rho}^*$ gm/cm^3	$\overline{M}^*$ gms/gm-mole	$\overline{I}^*$ cm^2/sec^2	$\overline{C}^*$ $gm\text{-}moles/cm^3$
10	Coaxial Entry in a Center Tube with Length=5 radii, Center Tube Velocity=20 cm/sec Annulus Velocity=0 cm/sec	Air	319	1.013×10^{6}	1.108×10^{-3}	29.00	2.286×10^{9}	-
11	Coaxial Parabolic Entry in a Long Tube. Length=20 Radii. Center tube max. velocity = 31.35 cm/sec with .005 moles/cm^3 H_2. Max. Annulus velocity = 27.95 cm/sec with .005 moles/cm^3 I_2	N_2, trace H_2 & I_2 N_2 is ref.	716	2.586×10^{9}	1.217	28.02	5.312×10^{9}	3.409×10^{1}
12	Coaxial Parabolic Entry in a Long tube. Length=5 radii. Center tube max velocity = 31.35 cm/sec with .005 moles/cm^3 H_2. Max. Annulus velocity =27.95 cm/sec with .005 moles/cm^3 I_2	N_2, trace H_2 & I_2 N_2 if ref.	716	2.586×10^{9}	1.217	28.02	5.312×10^{9}	3.409×10^{1}
13	Coaxial Entry in a tube. Length =5 radii. Center tube velocity= 20 cm/sec with air. Annulus velocity=10 cm/sec with temperature=1300°F, composition CO=.1, O_2=.1, CO_2=.1, H_2O=.12, N_2=.58	Air is Ref.	530	1.013×10^{6}	1.195×10^{-3}	28.85	2.126×10^{9}	4.122×10^{-5}

Table 1 - Description of Computations (Continued)

Problem Type #	Δt	ΔR	$\bar{R}_W$ cm	$\bar{V}$ cm/sec	$\bar{t}_{\sim}$ sec	N_{Re} (1)	N_{Eu}	N_{S1}	Grid NRxNL	R_A	$\bar{L}_T$ cm	Figure #	Θ	Φ	β_M	β_V	β_{MK}
1	.002	.25	.25	31.24 (2)	1	205	990	.008	4x4	.25	1	4	1	1	0	0	0
	.0015	.1	.25	31.24 (2)	1	205	990		10x4	.1		4					
	.001	.04	.25	31.24 (2)	1	205	990		25x4	.04		4					
2	.025	.2	1	34294 (3)	.001	34700	.714	.029	5x50	.2	50	5a,b, c,d, 6,7,8	1	1	0	0	0
	.01	.2	1	34294	.001	34700	.714	.029	5x50	.2	50	6	1	1	0	0	0
	.025	.2	1	34294	.001	34700	.714	.029	5x50	.2	50	8	.5	.5	0	0	0
	.025	.2	1	34294	.001	34700	.714	.029	5x50	.2	50	9	.5	.5	1	0	0
3	.00139	.05	.5	17.45	1	50	3.05×10^6	.029	20x5	.05	2.5	10	1	1	0	0	0
	.00417	.05	.5	17.45	.09	50	3.05×10^6	.318	20x5	.05	2.5	11	1	1	0	0	0
4	.00417	.05	.5	17.45	.36	50	3.05×10^6	.080	20x5	.05	2.5	12,13	1	1	0	0	0
5	Start .005, →.02	.10	1	10	1	57.3	9.14×10^6	.100	10x30	.15	20	14,15, 16,17, 18	1	1	0	0	0
6	.002 →.01	.05	2	12.5	1	143	5.58×10^6	.160	20x20	1	2	19→23	1	1	0	1.01	0
7	.002 →.01	.05	2	17.5	1	201	2.99×10^6	.115	20x20	1	2	24,25	1	1	0	1.5	0
8	.002 →.02	.05	2	12.5	1	143	5.58×10^6	.160	20x20	.05	40	26,27	1	1	0	1.1	0
9	.002 →.02	.05	2	17.5	1	201	2.99×10^6	.115	20x20	.05	40	28	1	1	0	1.2	0
10	.002 →.01	.05	2	5	1	57.3	3.66×10^7	.4	20x20	.2	10	29a,b, c,g,h, 30b	1	1	0	1.01	0
10	.002 →.01	.05	2	5	1	57.3	3.66×10^7	.4	20x20	.2	10	29d,e, f,30a	1	1	1.05	1	0

Table 1 - Description of Computations (Continued)

Problem Type #	Δt	ΔR	$\overline{R}_W$ cm	$\overline{V}$ cm/sec	$\overline{t}_{\sim}$ sec	N_{Re} (1)	N_{Eu}	N_{S1}	Grid NRxNL	R_A	$\overline{L}_T$ cm	Figure #	Θ	Φ	β_M	β_V	β_{MK}
11	.10	.05	10.06	17.18	1	248	7.2×10^6	.586	20x20	.05	201.20	31,32, 33,34	1	1	0	1.01	1.01
12	.05	.05	10.06	17.18	1	248	7.2×10^6	.586	20x20	.2	50.12	35,36	1	1	0	1.01	1.01
13	.01	.05	2	12.5	1	160.6	5.43×10^6	.160	20x20	.2	10	37	1	1	.99	.99	.99

(1) This is based on the radius and mean velocity.

(2) Centerline velocity used only for this case. Mass mean velocity used everywhere else.

(3) This is the speed of sound. Dimensionless velocity then is Mach Number

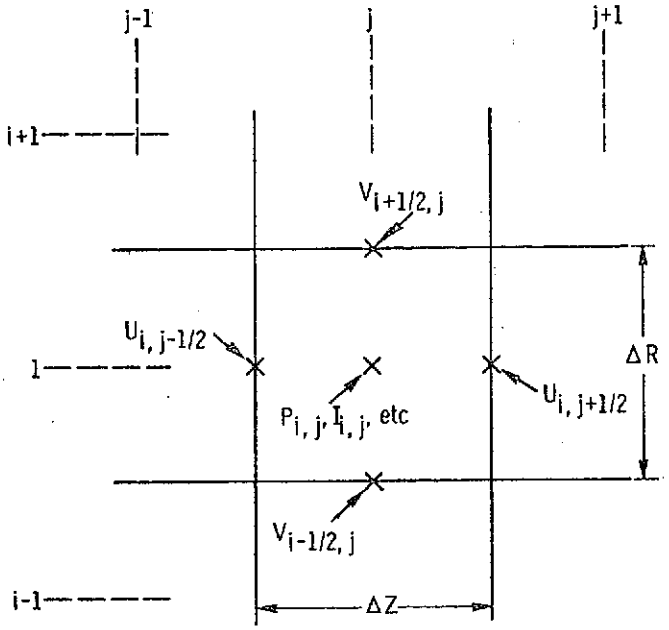


Figure 1. - A typical ICE cell showing the variable locations.

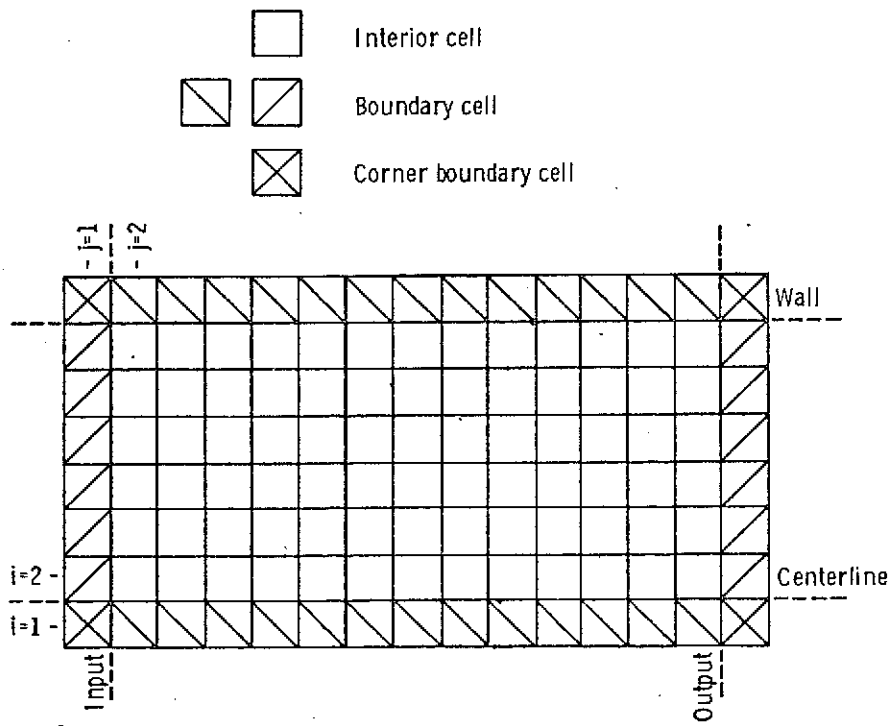


Figure 2. - ICE computational grid.

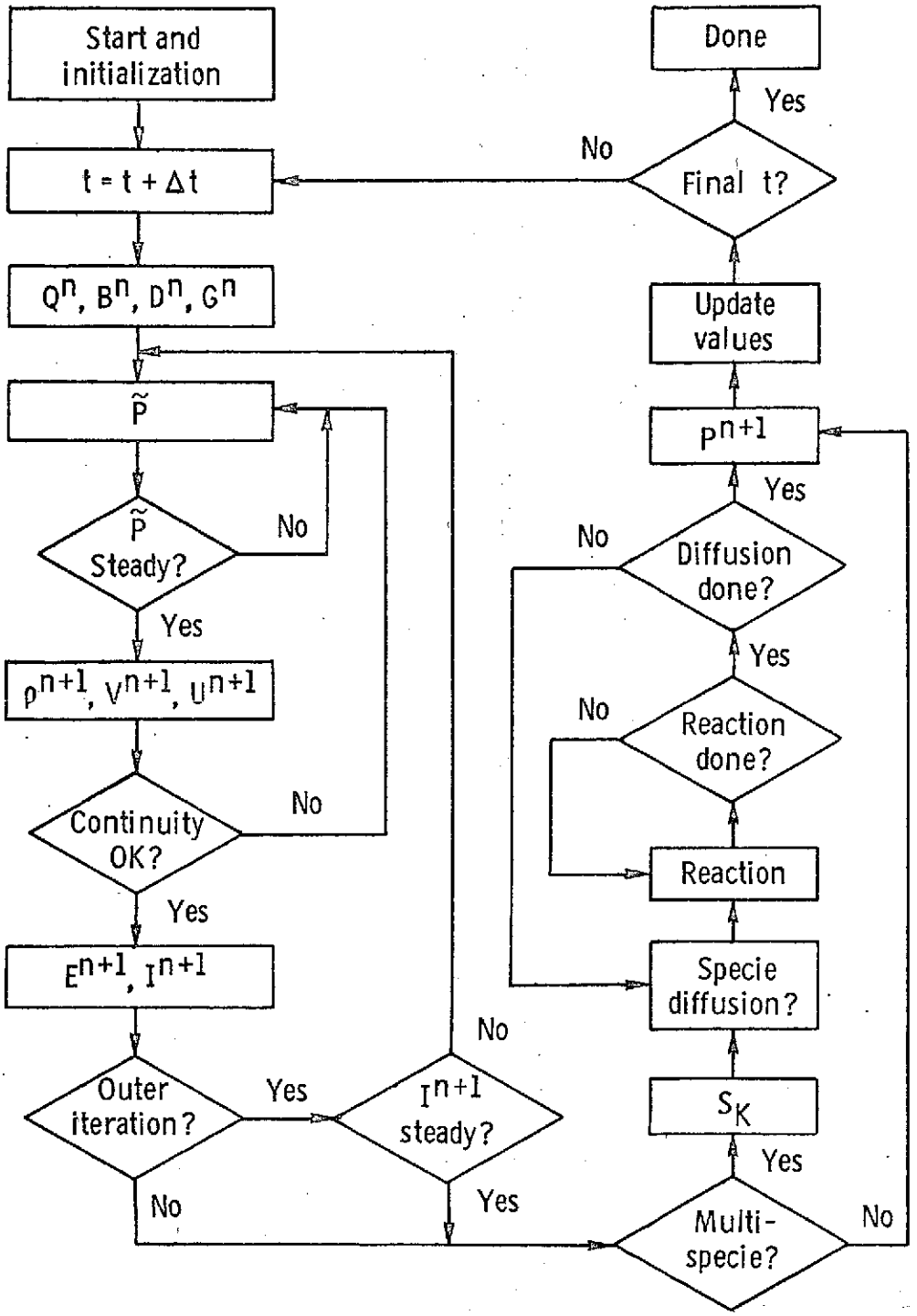


Figure 3. - Basic flow sheet for the ICE algorithm.

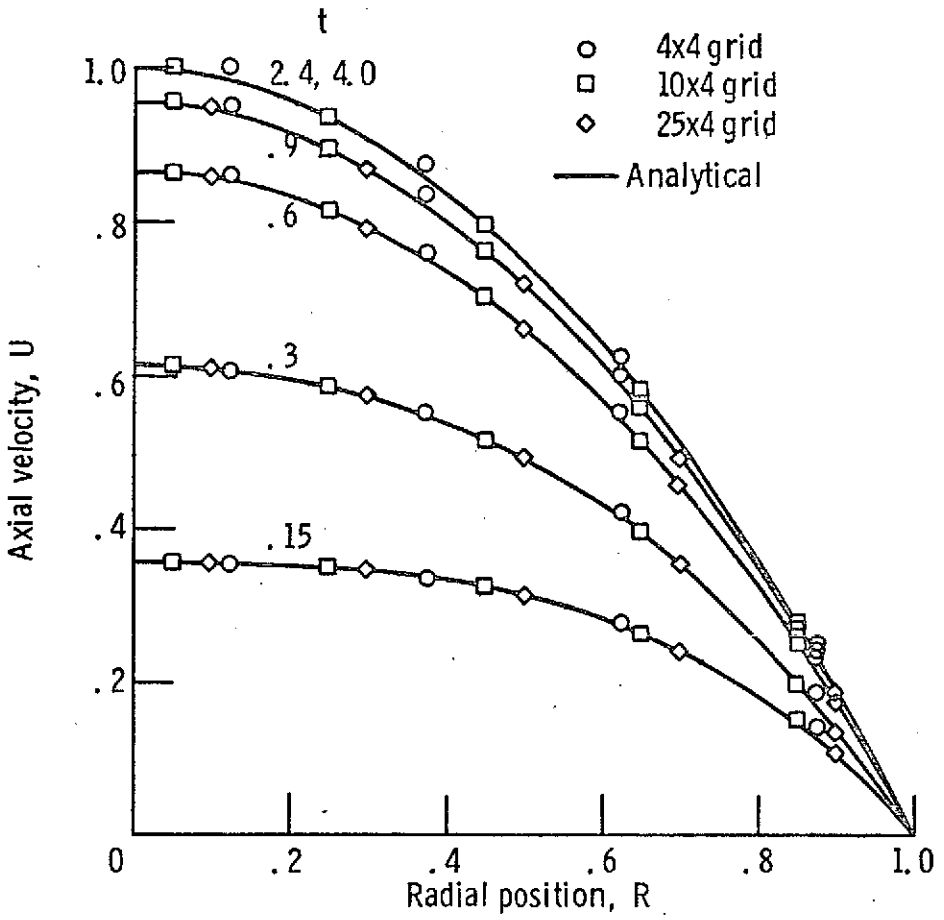


Figure 4. - Startup of incompressible flow in an infinite tube. Re = 205 at steady state.

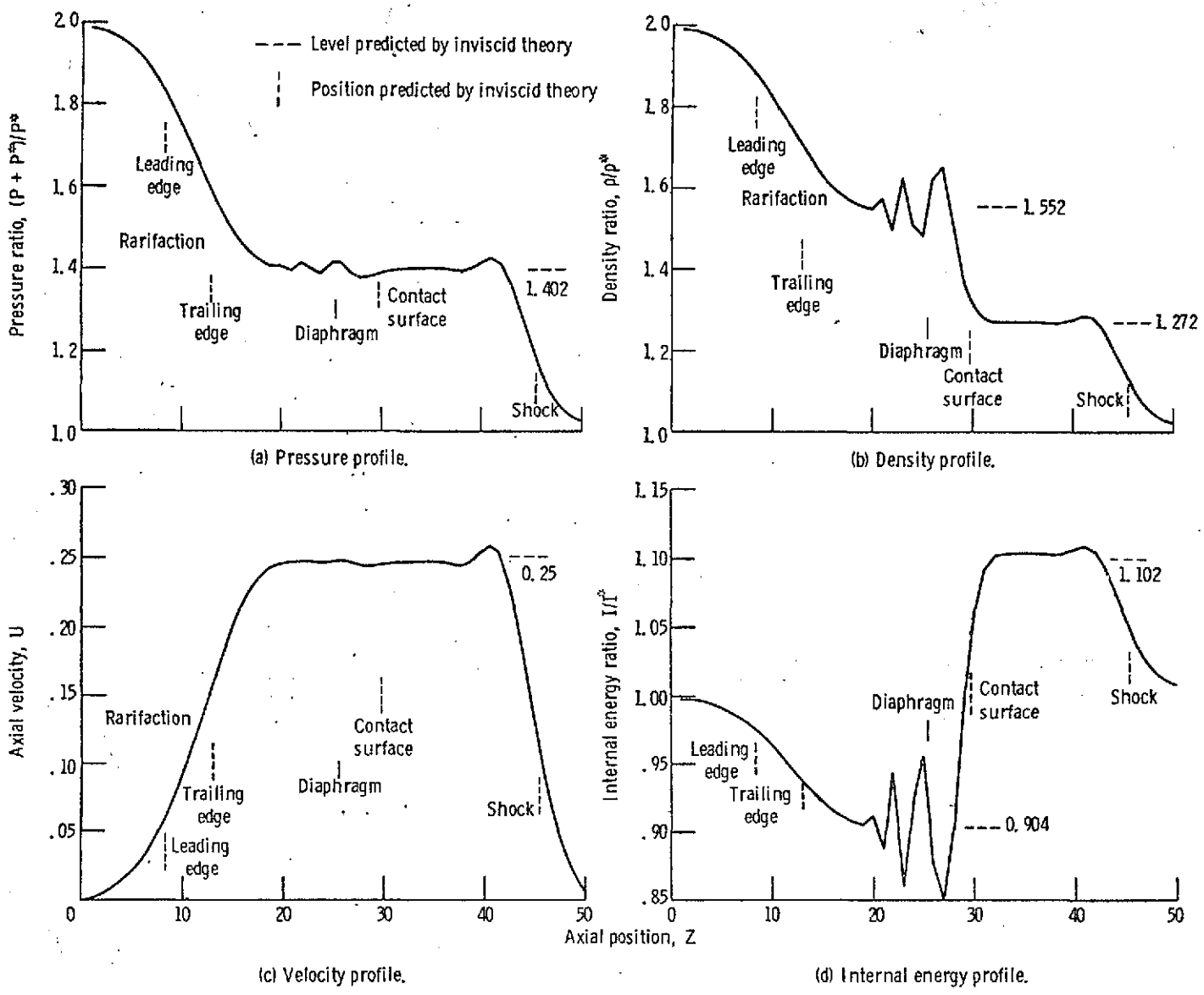


Figure 5. - Profile for the shock tube problem, t = 0.5.

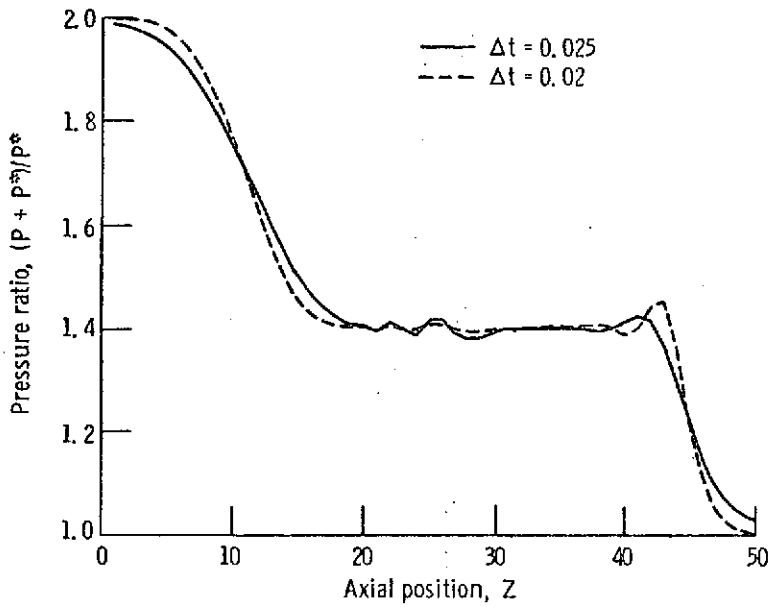


Figure 6. - Pressure profiles for the shock tube problem with two time step sizes, t = 0.5.

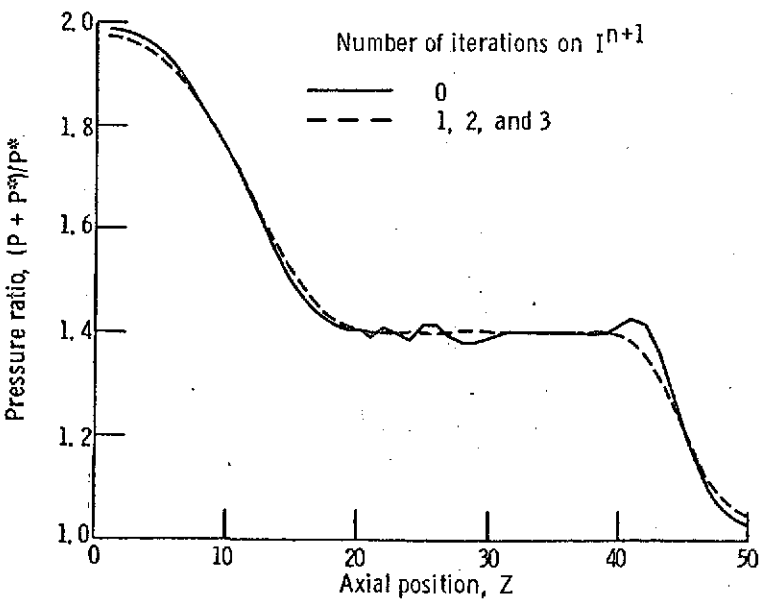


Figure 7. - Pressure profiles for the shock tube problem using I^{n+1} in the hybrid function, t = 0.5.

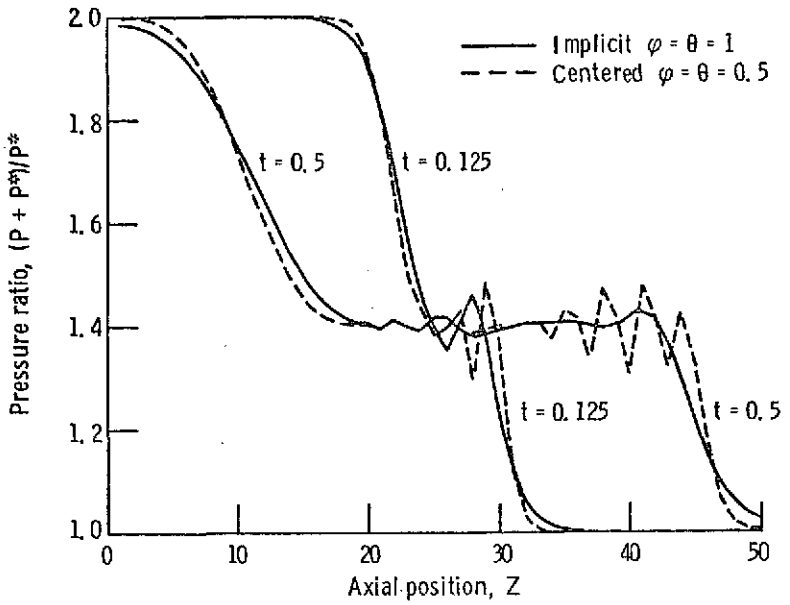


Figure 8. - Effect of time centering on pressure profiles for the shock tube problem, t = 0.125 and 0.5.

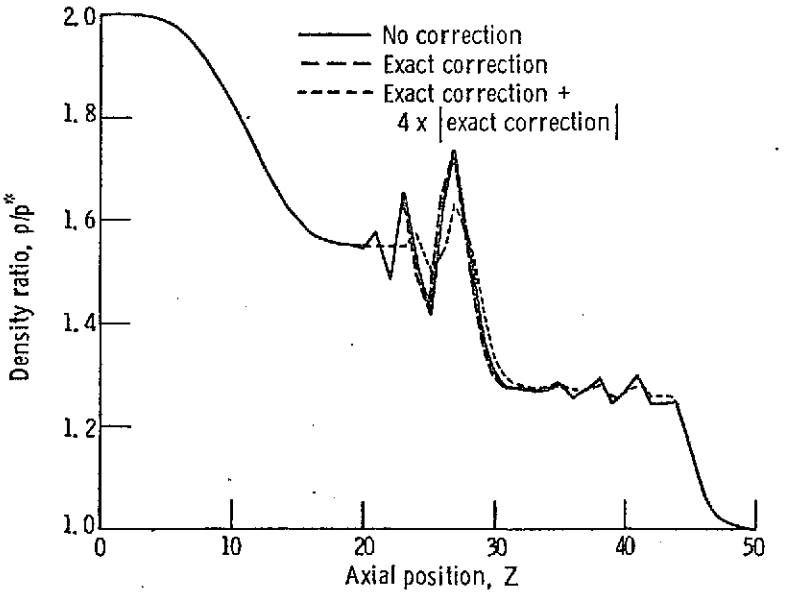


Figure 9. - Effect of the mass truncation error correction, β_M, on a density profile for the shock tube problem. The calculation is time-centered, $\theta = \varphi = 0.5$.

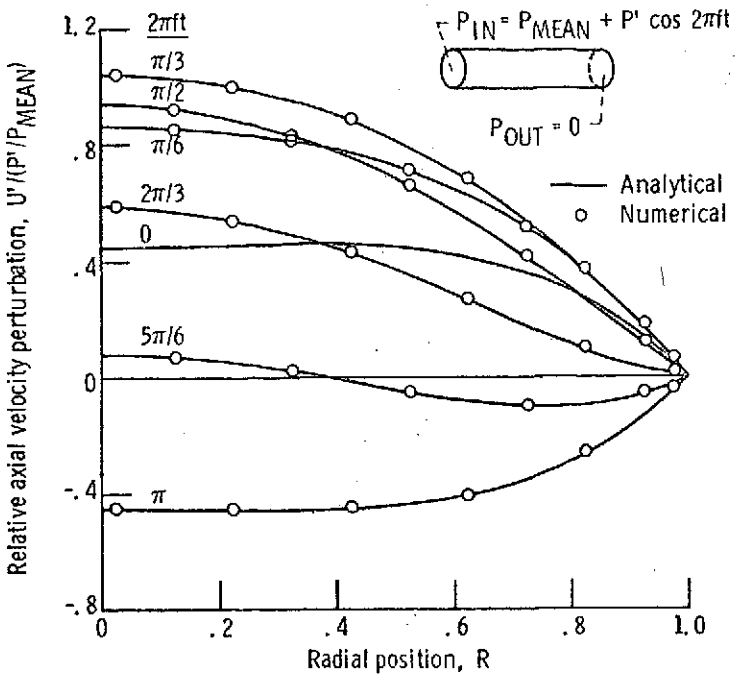


Figure 10. - Radial distributions of the axial velocity perturbations for oscillatory incompressible tube flow, $\sqrt{2\pi f/\nu}\, R_W = 3$.

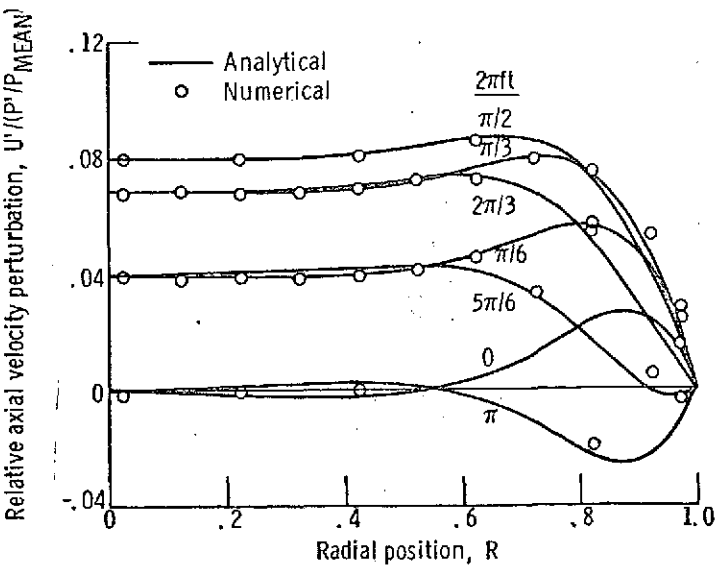


Figure 11. - Radial distributions of the axial velocity perturbations. For oscillatory incompressible tube flow, $\sqrt{2\pi f/\nu}\, R = 10$.

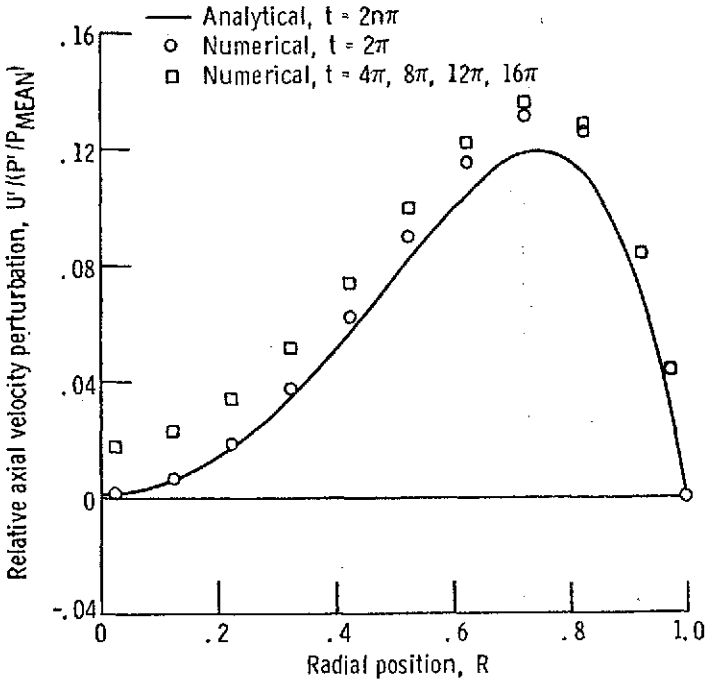


Figure 12. - Radial distributions of axial velocity perturbations for startup of oscillatory incompressible tube flow, $\sqrt{2\pi f/\nu}\ R = 5$.

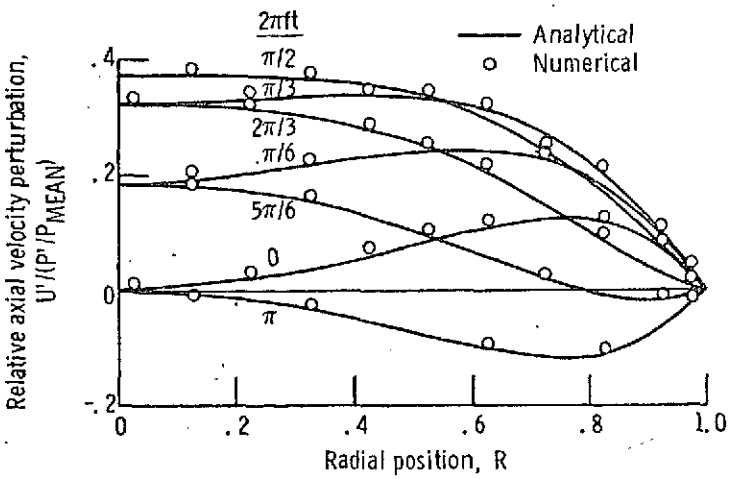


Figure 13. - Cyclic radial distributions of axial velocity perturbations for transient oscillatory incompressible tube flow, $\sqrt{2\pi f/\nu}\ R = 5$.

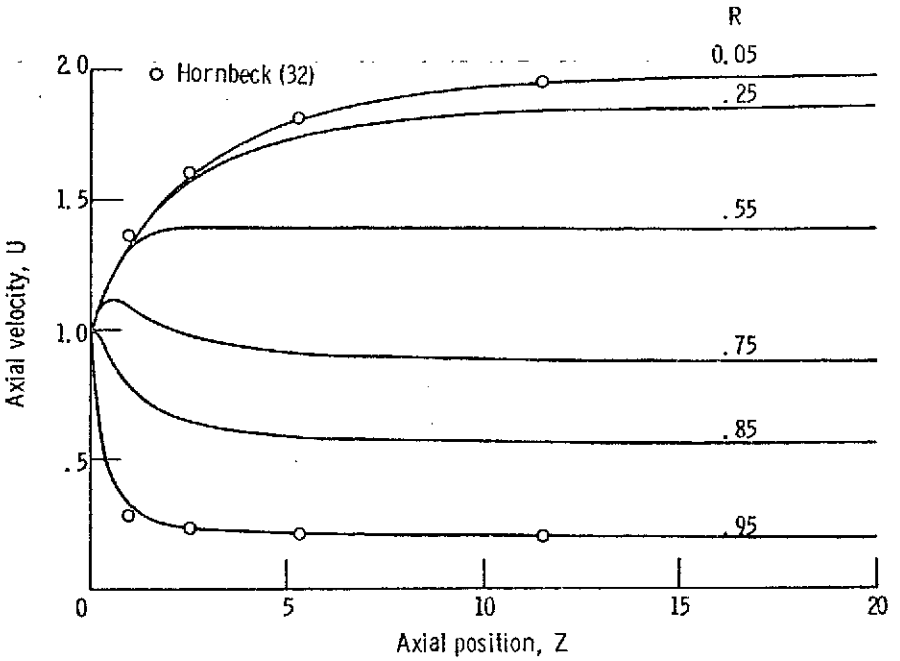


Figure 14. - Axial velocity development at several radial positions for laminar incompressible tube entry flow, t = 2.

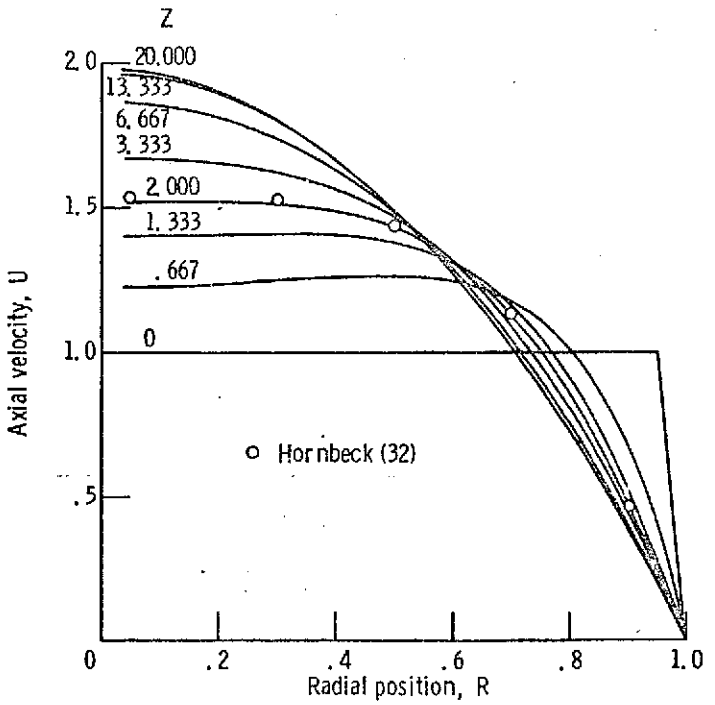


Figure 15. - Radial profiles of axial velocity at several downstream positions for laminar incompressible tube entry flow, t = 2.

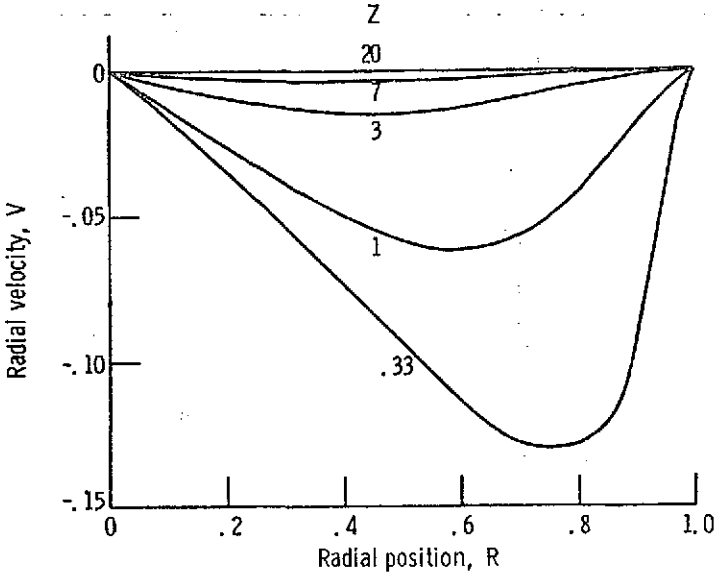


Figure 16. - Radial profiles of radial velocity at several downstream positions for laminar incompressible tube entry flow, t = 2.

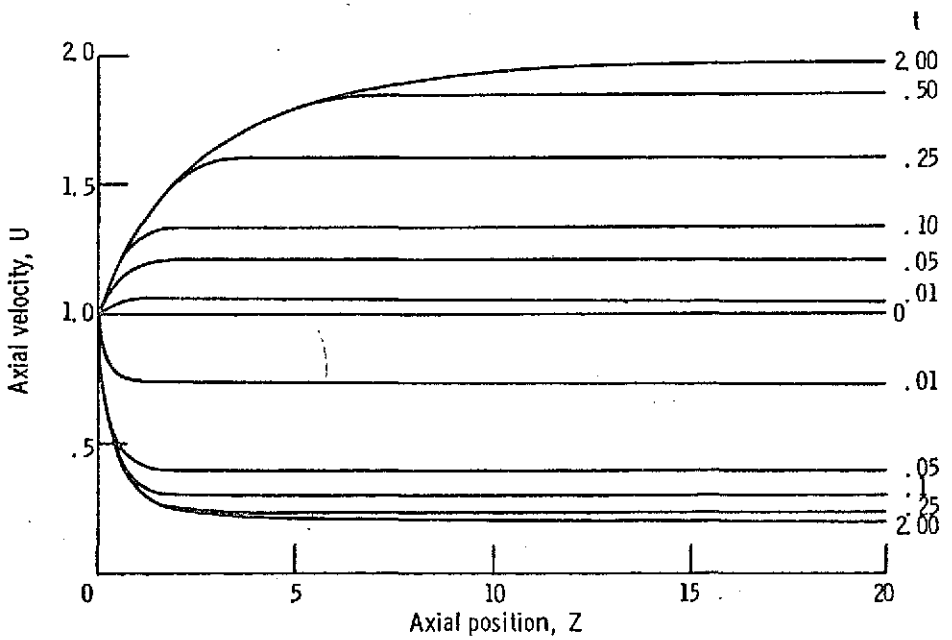


Figure 17. - Axial profiles of axial velocity during startup of laminar incompressible tube entry flow. Values > 1 for R = 0.05; values < 1 for R = 0.95.

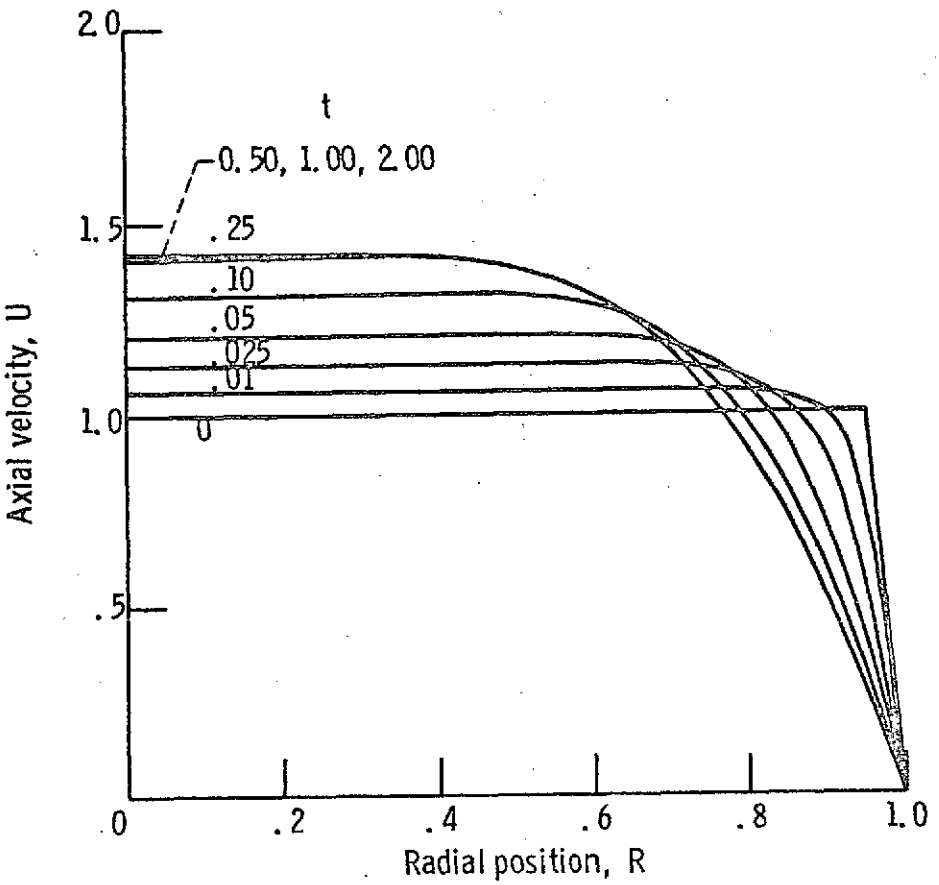


Figure 18. - Radial profiles of axial velocity during startup of laminar incompressible tube entry flow, Z = 1.333.

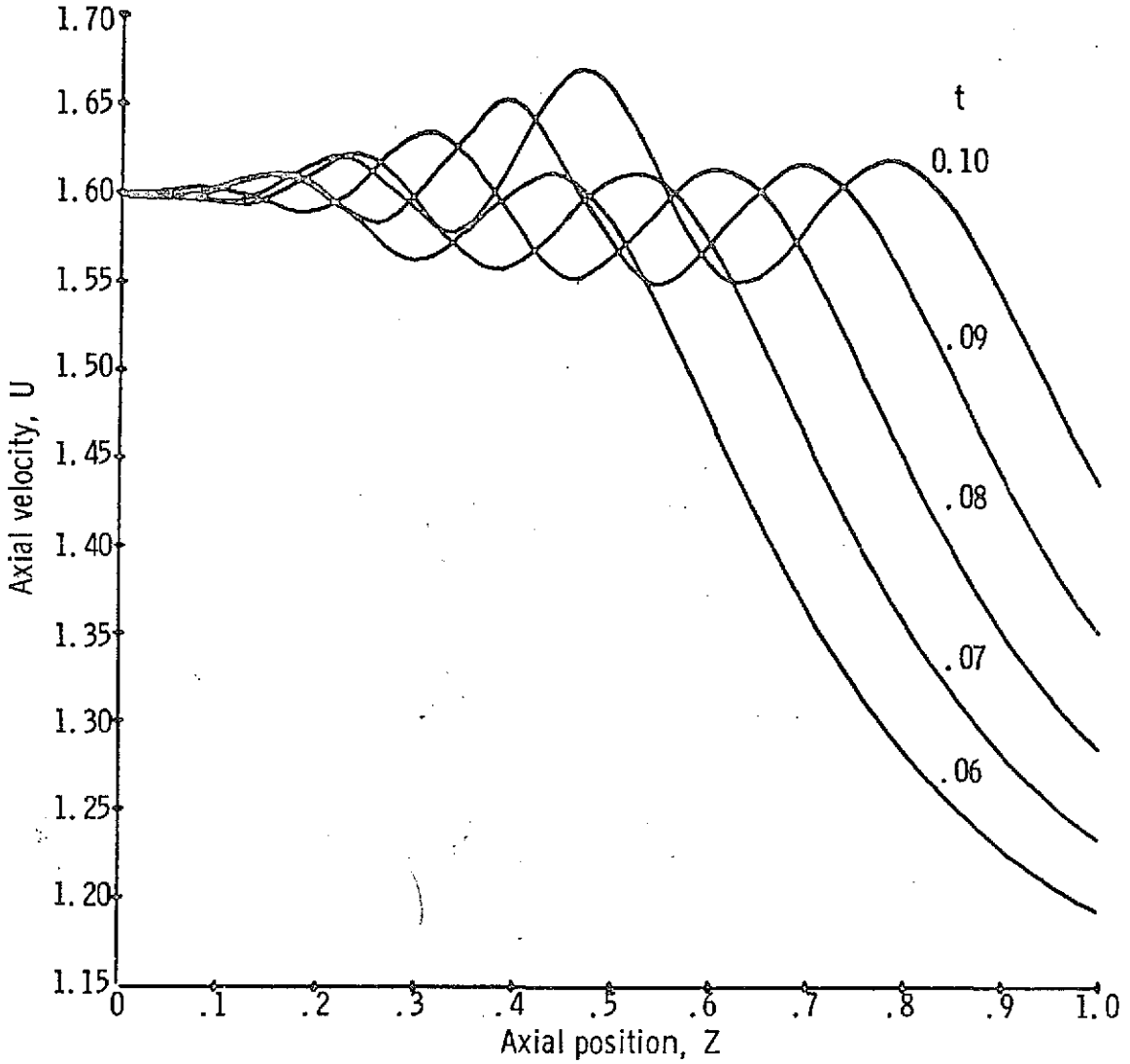


Figure 19. - Growth of instability of axial velocity for coaxial entry into a short tube. R = 0.025, entry velocity ratio = 2.0.

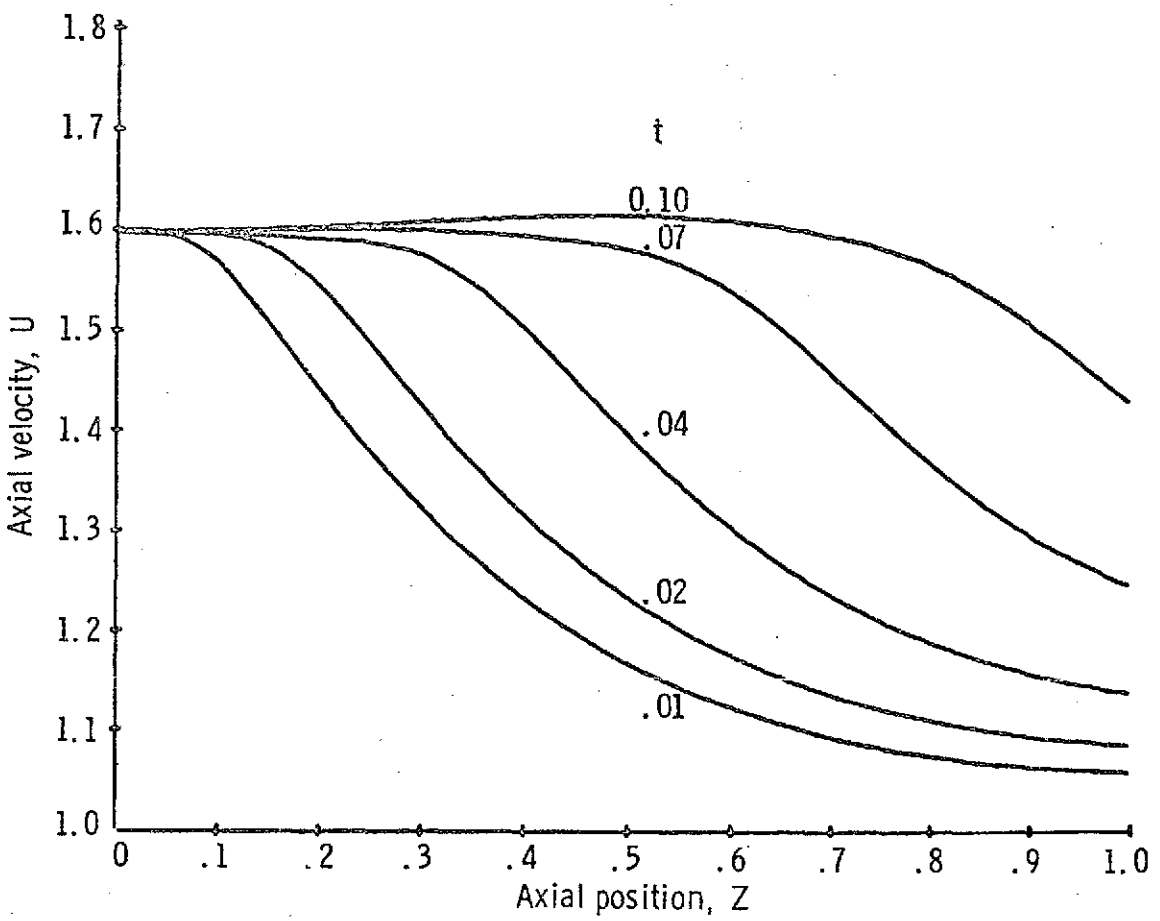
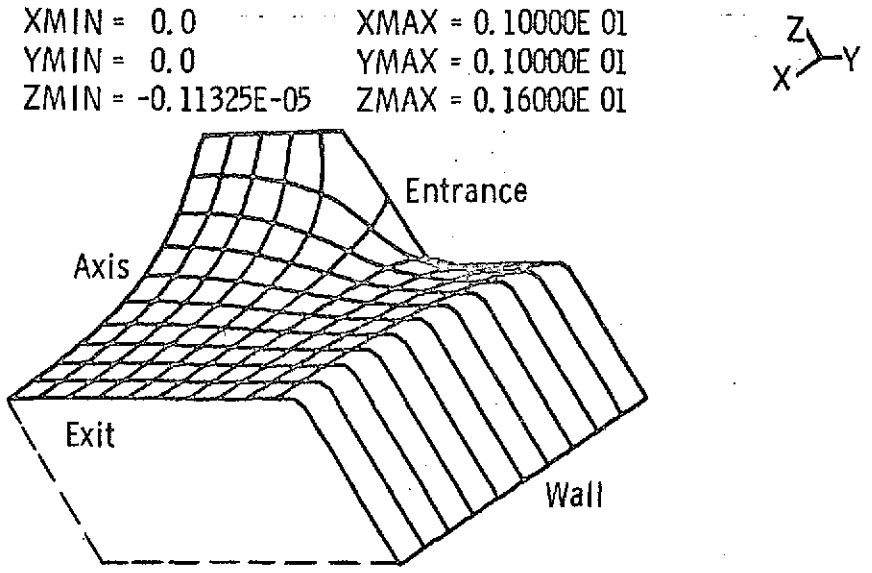
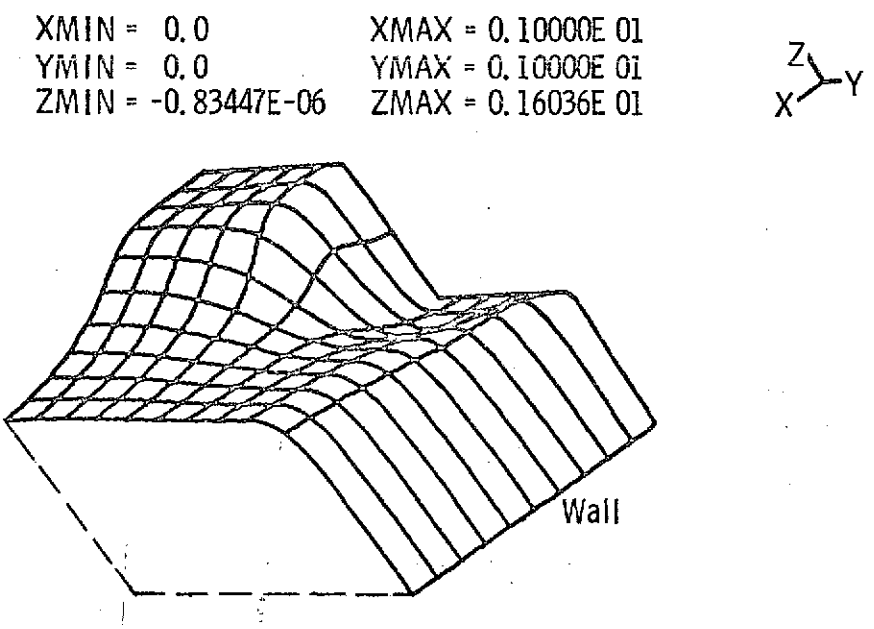


Figure 20. - Stabilized axial velocity for coaxial entry into a short tube R = 0.025, entry velocity ratio = 2.0.



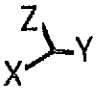
(a) t = 0.002.

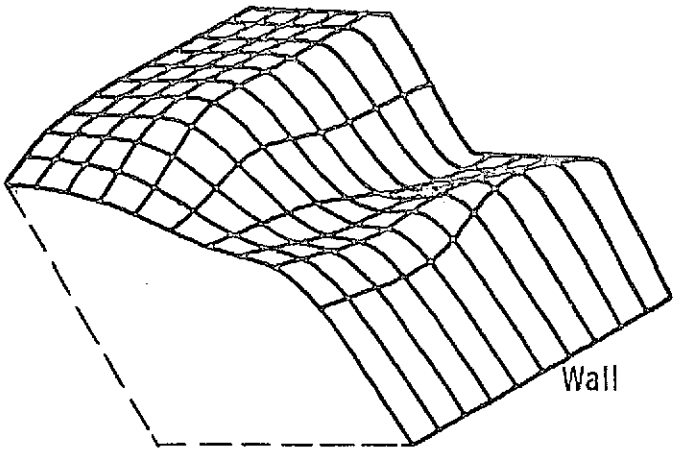


(b) t = 0.04.

Figure 21. - Plots of transient axial velocity for coaxial entry into a short tube. Entry velocity ratio = 2.0.

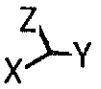
XMIN = 0.0 XMAX = 0.10000E 01
YMIN = 0.0 YMAX = 0.10000E 01
ZMIN = -0.65565E-06 ZMAX = 0.16319E 01

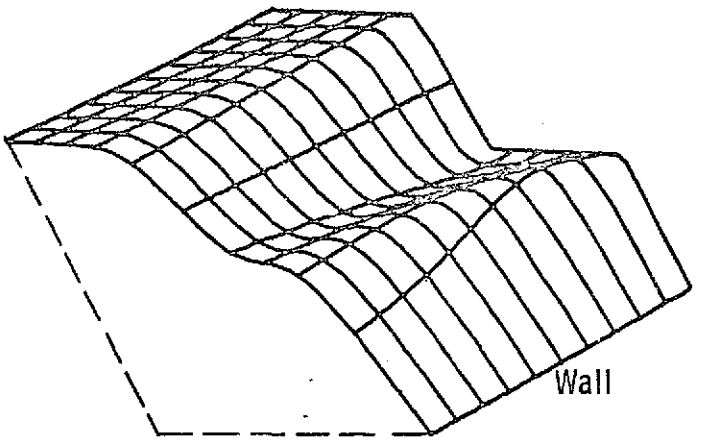




(c) t = 0.10.

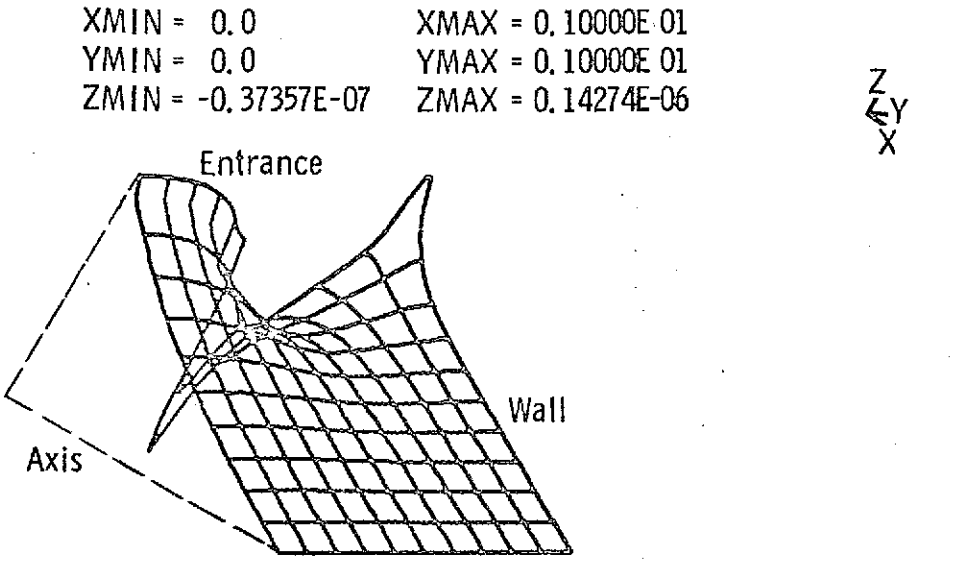
XMIN = 0.0 XMAX = 0.10000E 01
YMIN = 0.0 YMAX = 0.10000E 01
ZMIN = -0.89407E-06 ZMAX = 0.16833E 01



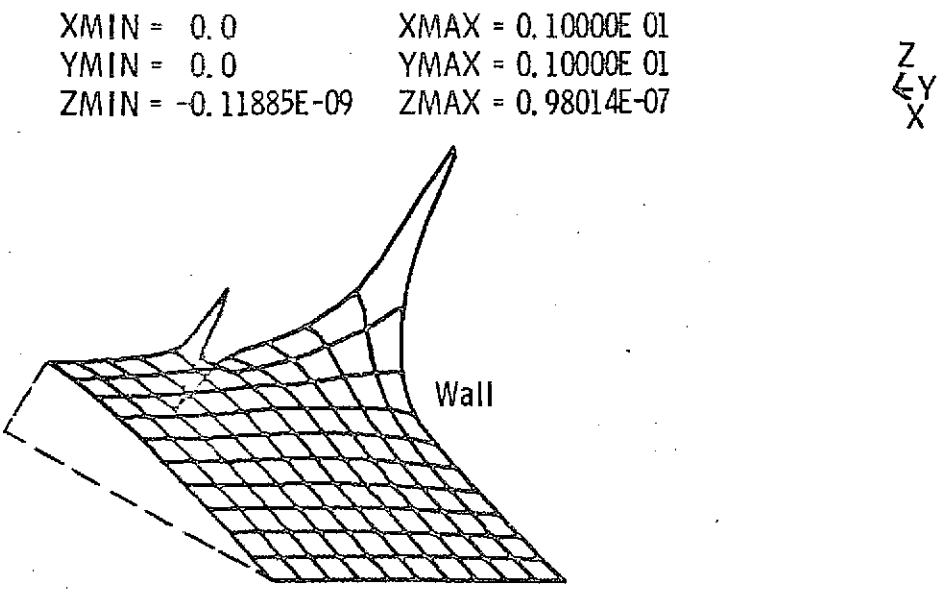


(d) t = 0.24.

Figure 21. - Concluded.

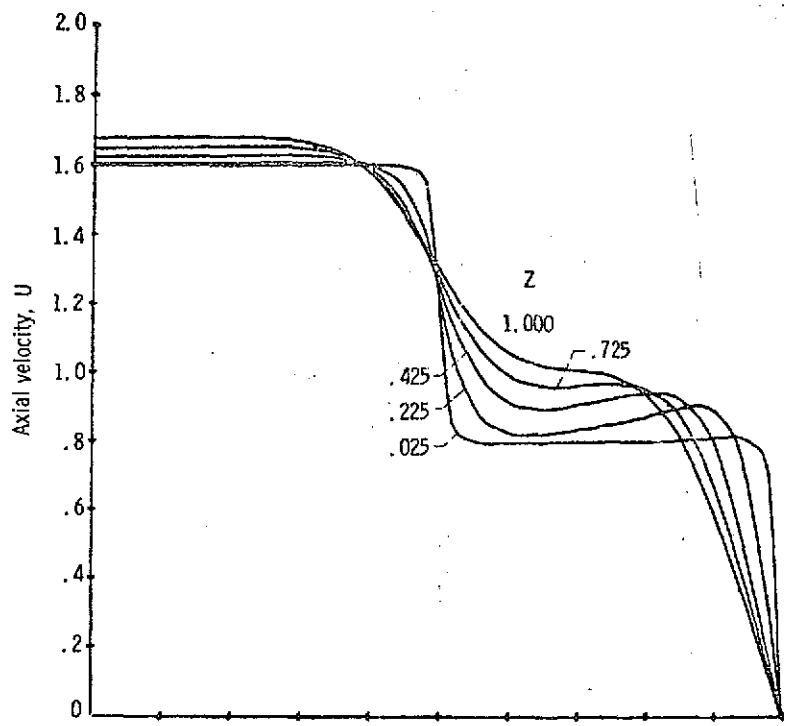


(a) t = 0.004.

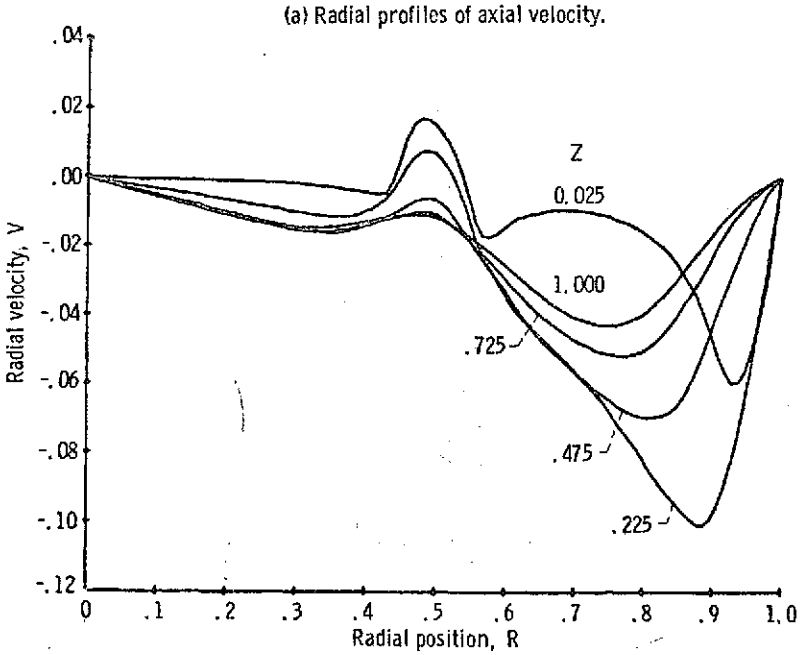


(b) t = 0.24.

Figure 22. - Plots of the transient pressure field for coaxial entry into a short tube. Entry velocity ratio = 2.0.

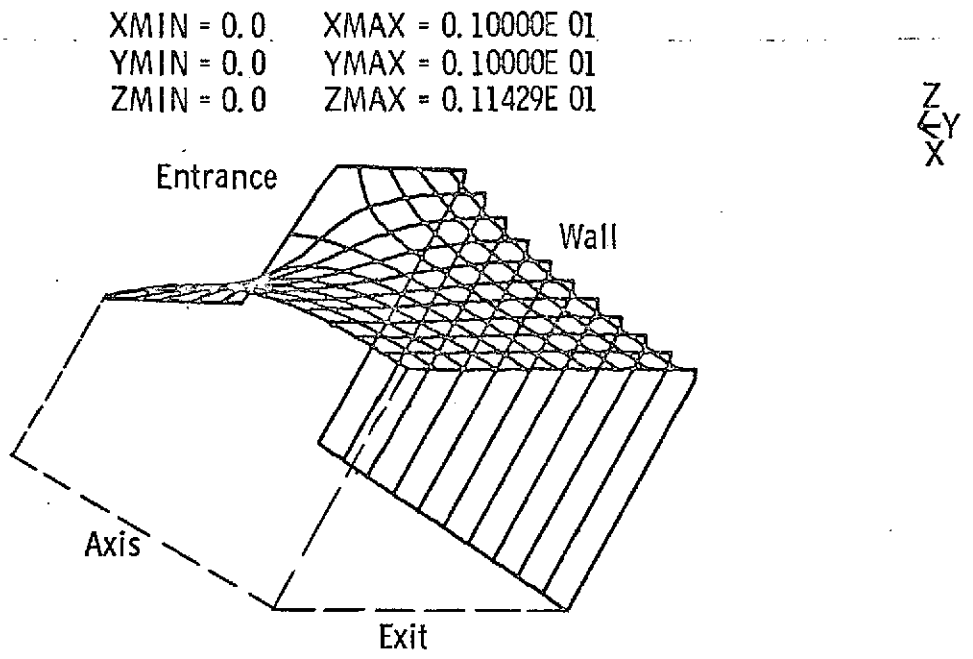


(a) Radial profiles of axial velocity.

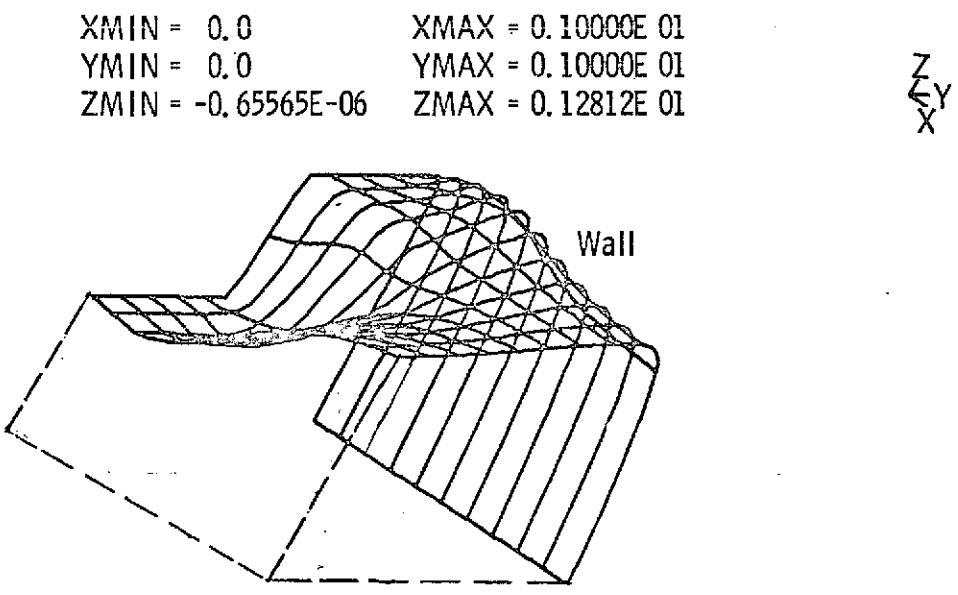


(b) Radial profiles of radial velocities.

Figure 23. - Coaxial entry into a short tube. t = 0.24 (steady state), entry velocity ratio = 2.0.



(a) t = 0.002.

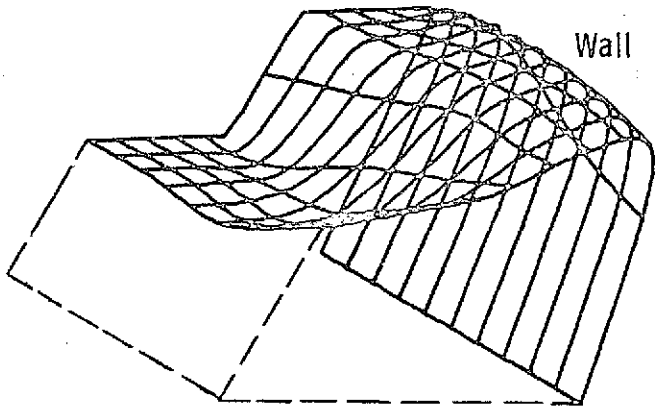


(b) t = 0.05.

Figure 24. - Plots of transient axial velocity for coaxial entry into a short tube. Entry velocity ratio = 0.05.

XMIN = 0.0 XMAX = 0.10000E 01
YMIN = 0.0 YMAX = 0.10000E 01
ZMIN = -0.59605E-06 ZMAX = 0.13113E 01

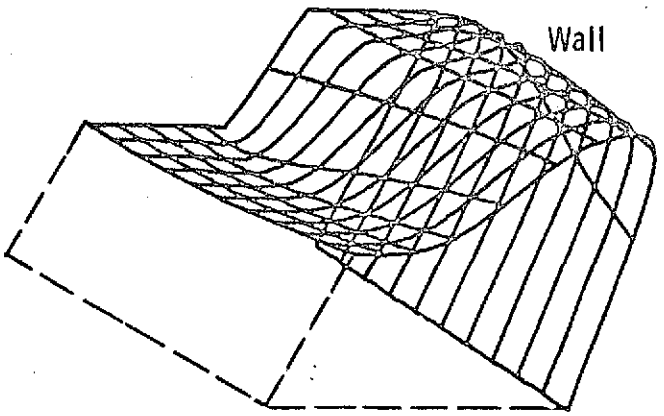
Z
Y
X



(c) t = 0.10.

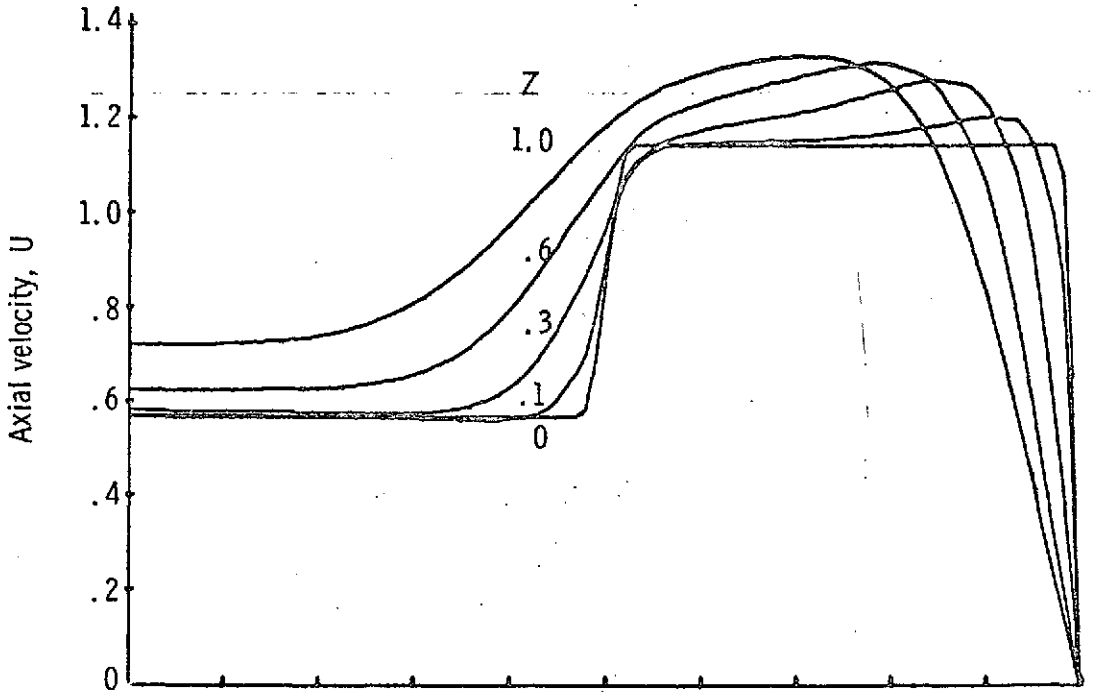
XMIN = 0.0 XMAX = 0.10000E 01
YMIN = 0.0 YMAX = 0.10000E 01
ZMIN = -0.10133E-05 ZMAX = 0.13325E 01

Z
Y
X

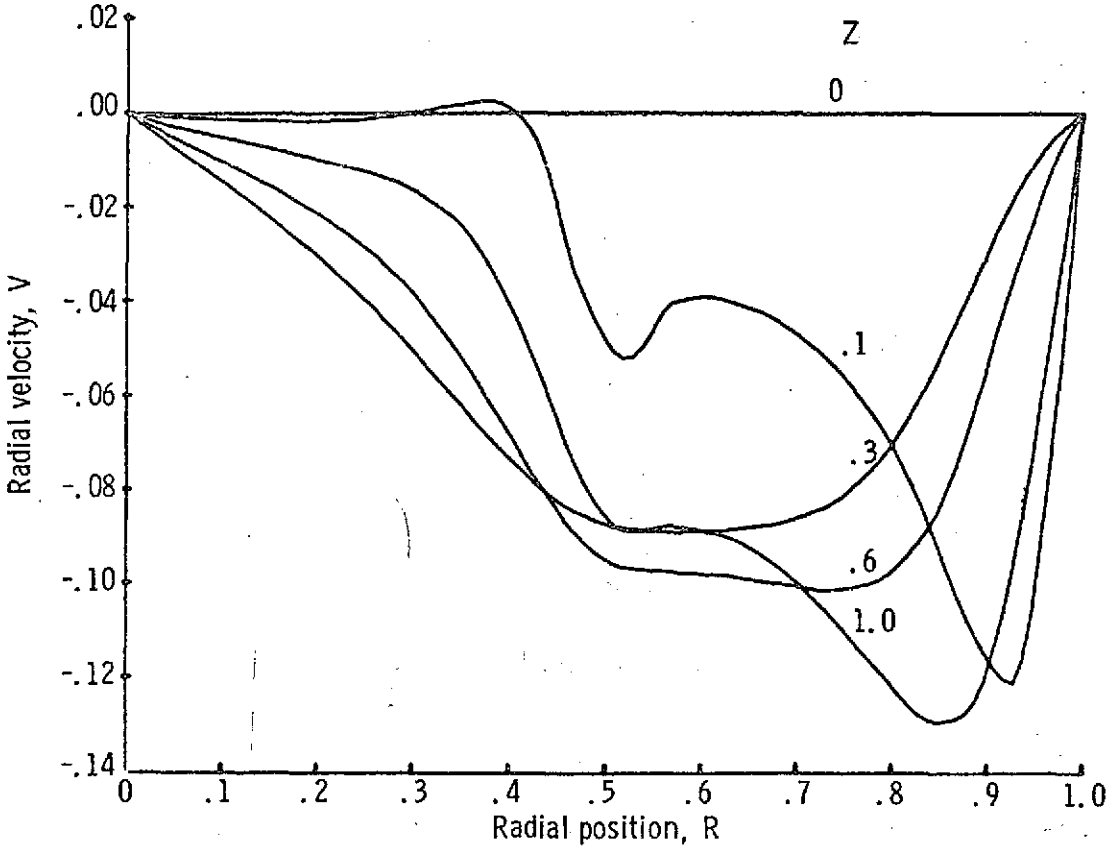


(d) t = 0.26

Figure 24. - Concluded.

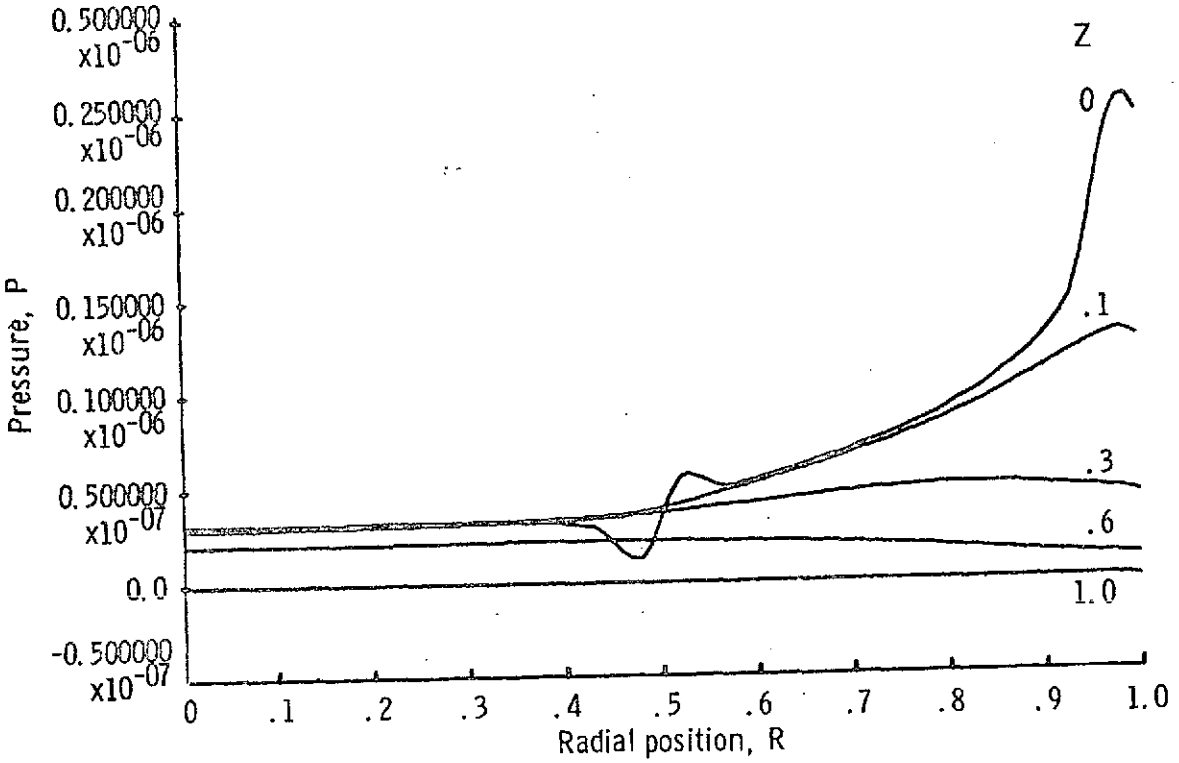


(a) Radial profiles of axial velocity.



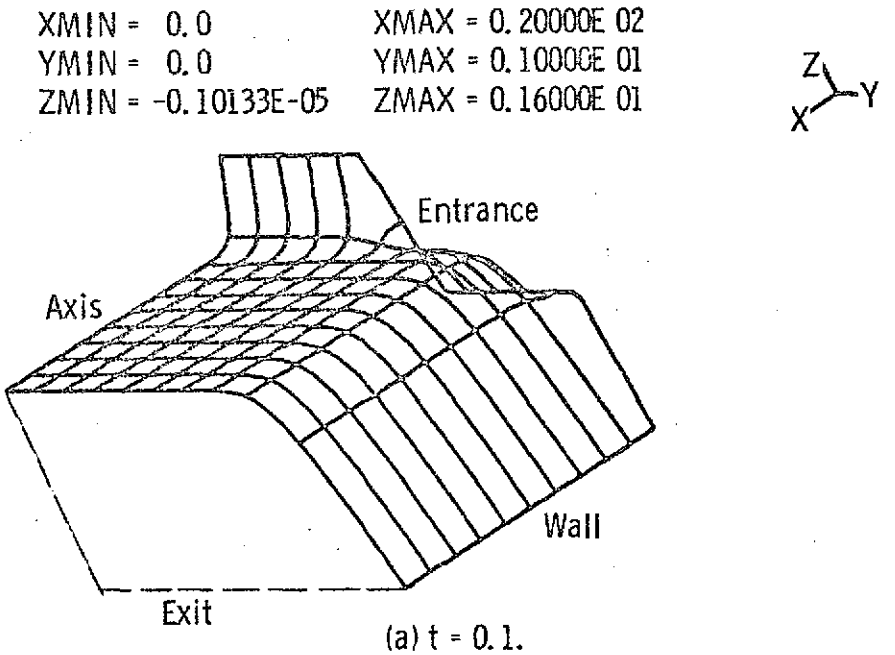
(b) Radial profiles of radial velocity.

Figure 25. - Coaxial entry into a short tube. t = 0.26 (steady state), entry velocity ratio = 0.05.

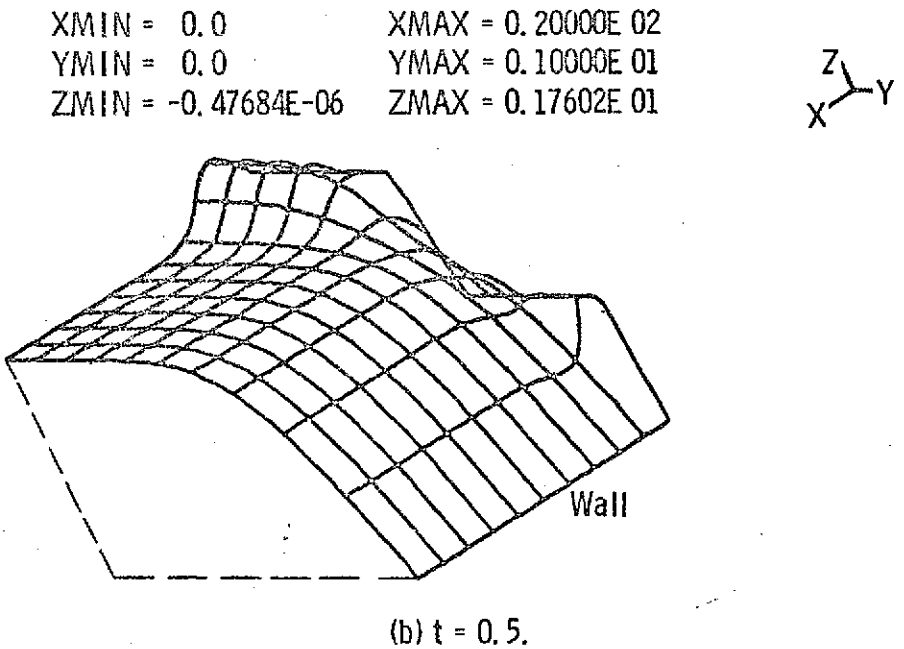


(c) Radial profiles of pressure.

Figure 25. - Concluded.



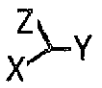
(a) t = 0.1.

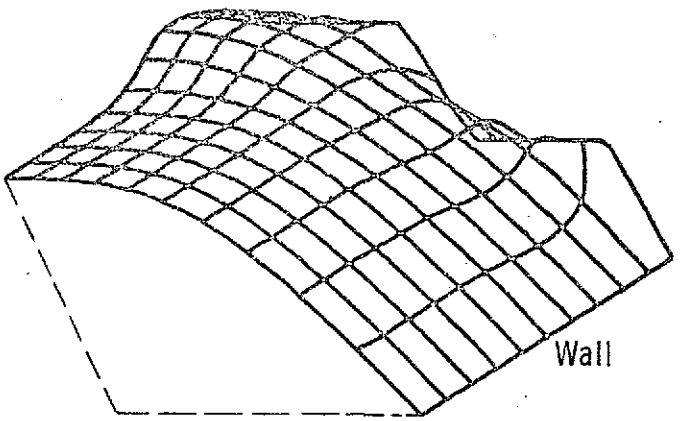


(b) t = 0.5.

Figure 26. - Plots of transient axial velocity for coaxial entry into a long tube. Entrance velocity = 2.0.

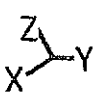
XMIN = 0.0 XMAX = 0.20000E 02
YMIN = 0.0 YMAX = 0.10000E 01
ZMIN = -0.23842E-06 ZMAX = 0.18747E 01

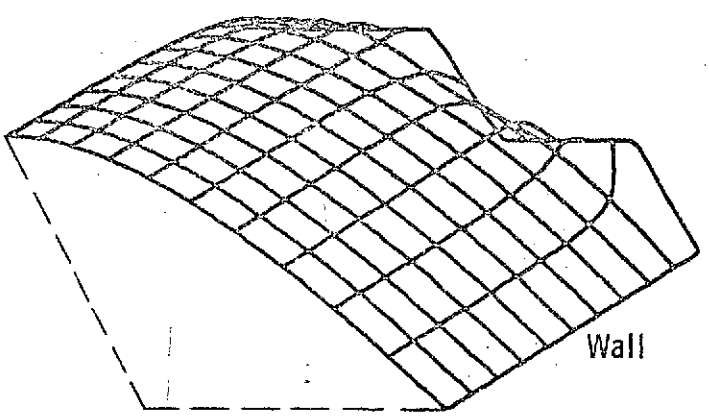




(c) t = 1.0.

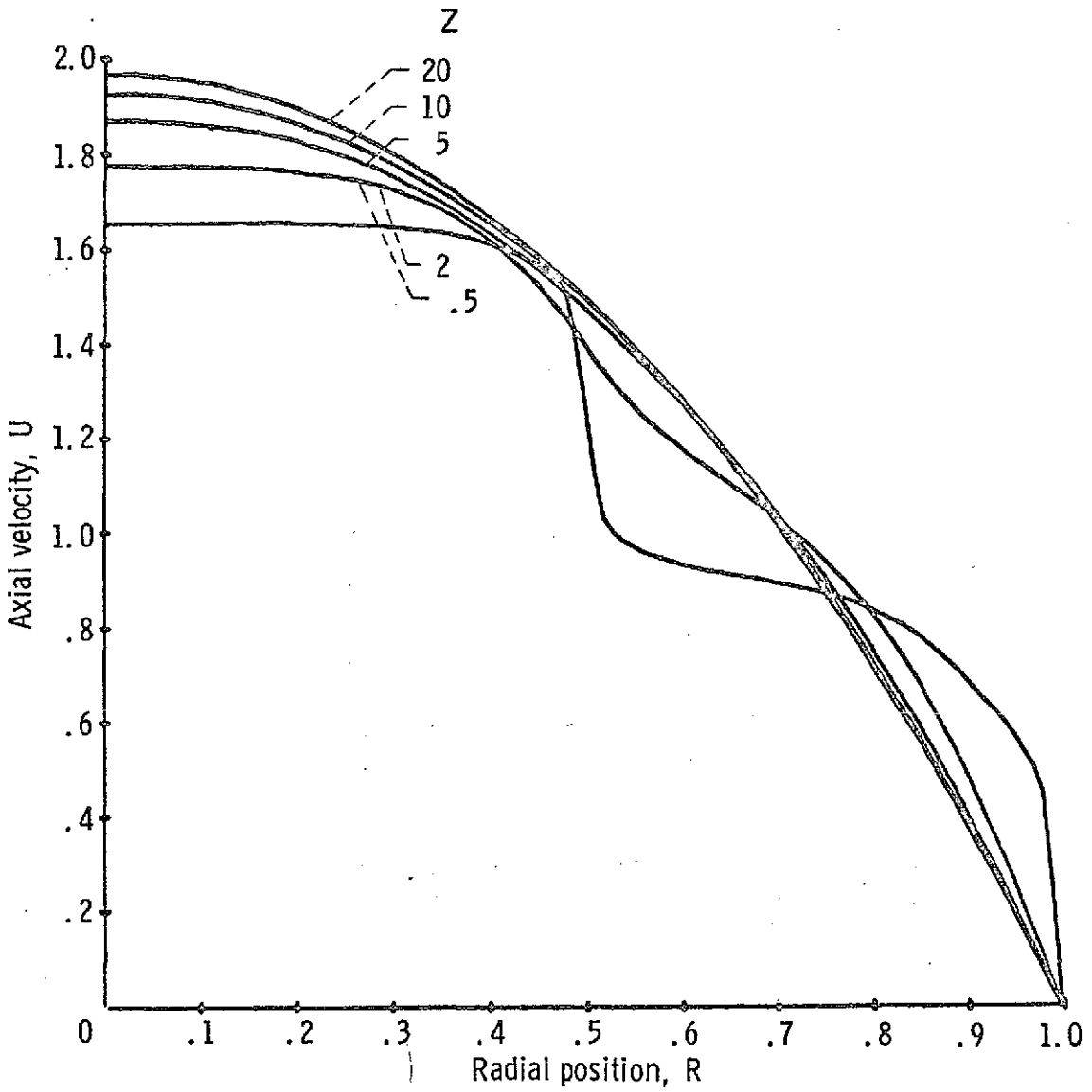
XMIN = 0.0 XMAX = 0.20000E 02
YMIN = 0.0 YMAX = 0.10000E 01
ZMIN = -0.23842E-06 ZMAX = 0.19741E 01





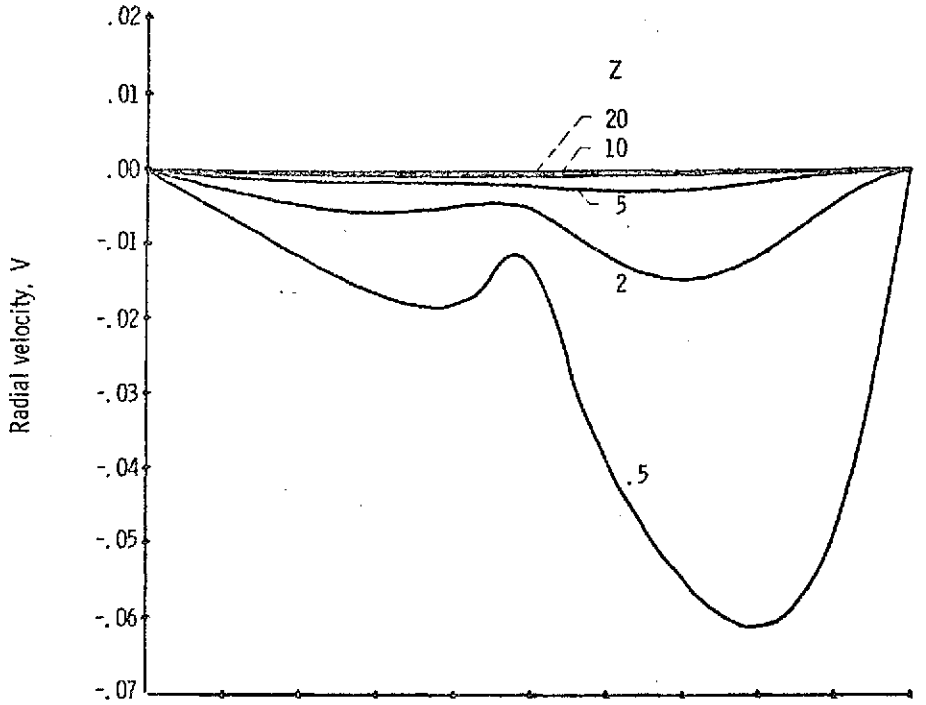
(d) t = 3.5.

Figure 26. - Concluded.

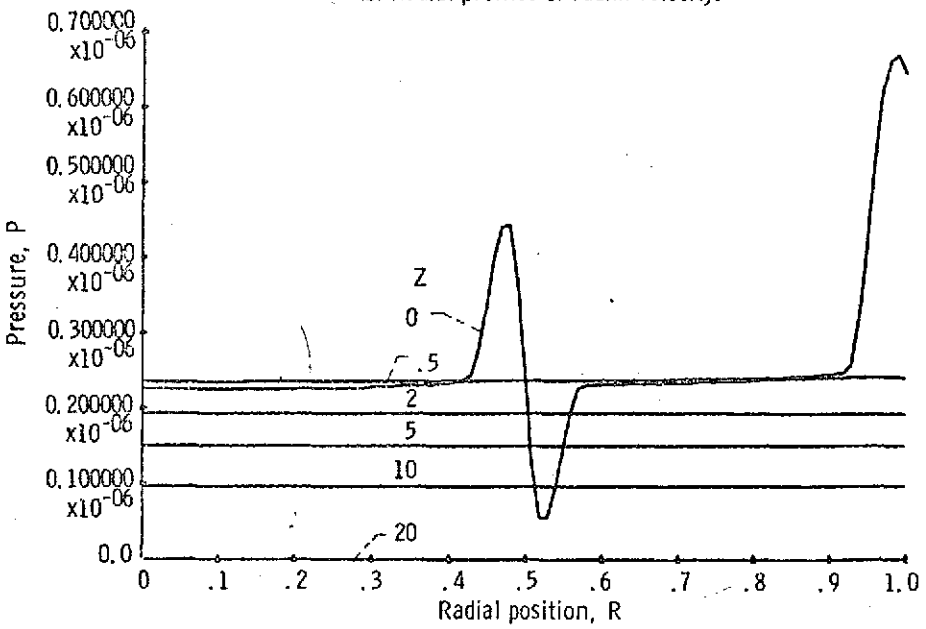


(a) Radial profiles of axial velocity.

Figure 27. - Coaxial entry into a long tube. t = 3.5 (steady state), entry velocity ratio = 2.0.

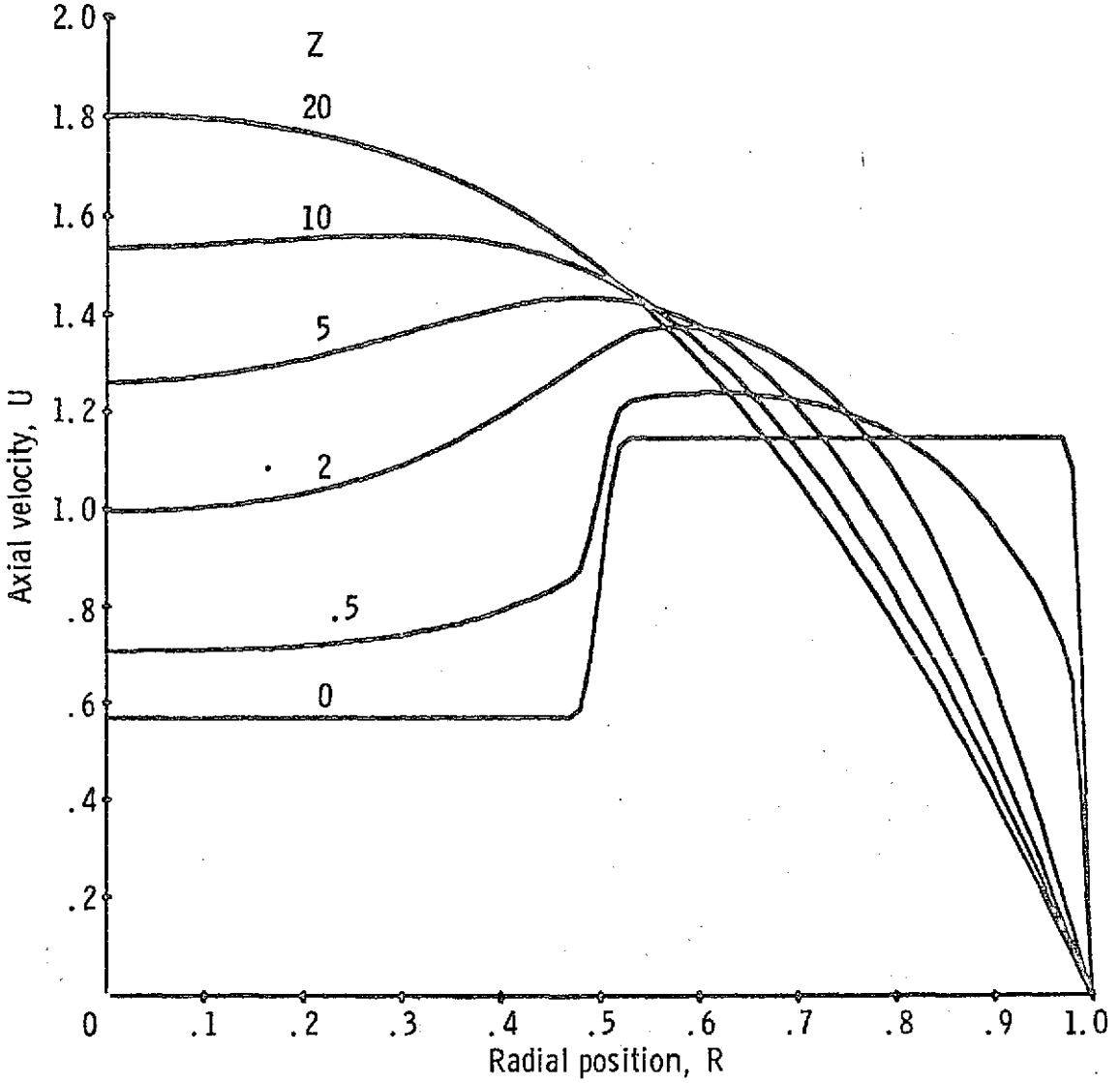


(b) Radial profiles of radial velocity.



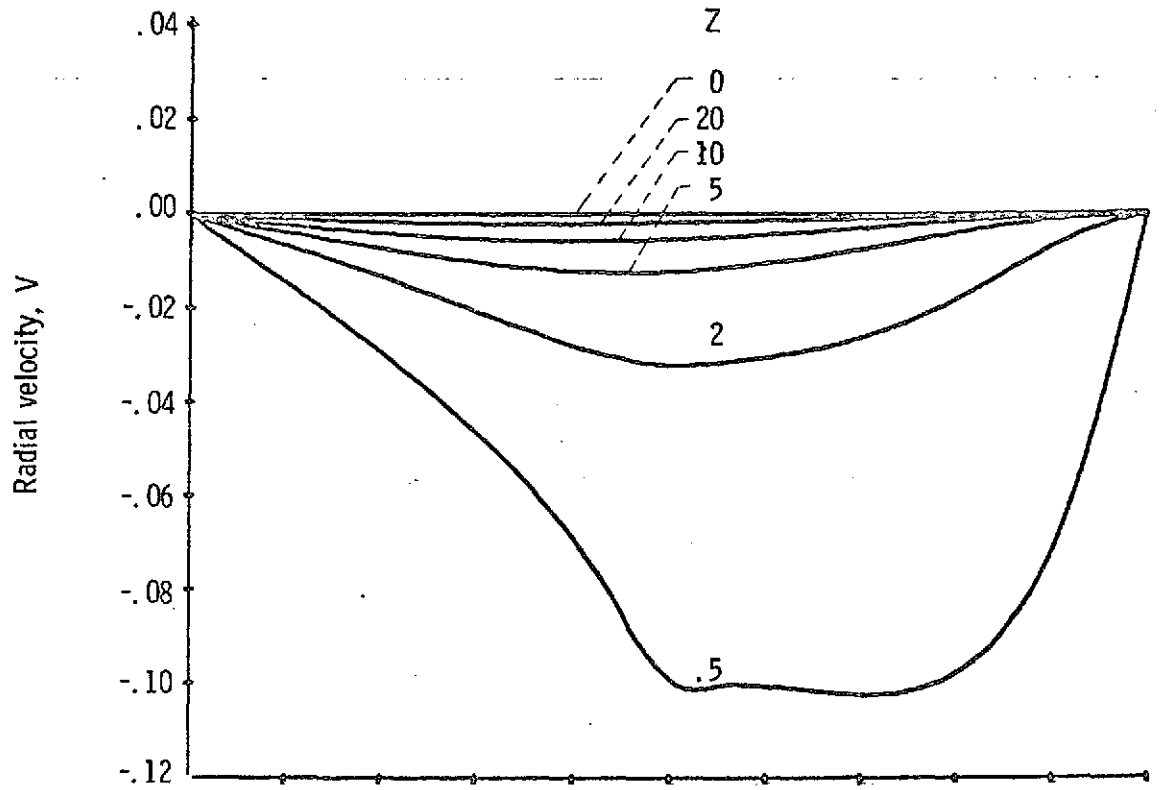
(c) Radial profiles of pressure.

Figure 27. - Concluded.

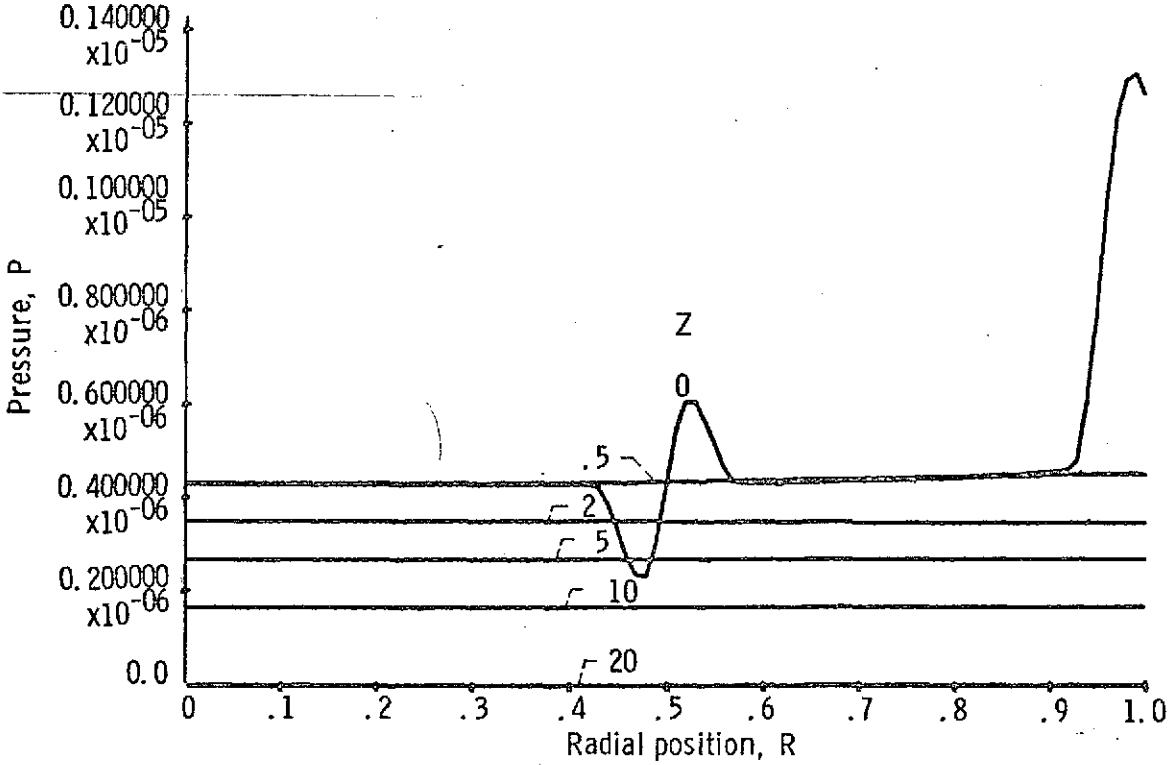


(a) Radial profiles of axial velocity.

Figure 28. - Coaxial entry into a long tube. t = 2.4 (steady state) entry velocity ratio = 0.05.



(b) Radial profiles of radial velocity.



(c) Radial profiles of pressure.

Figure 28. - Concluded.

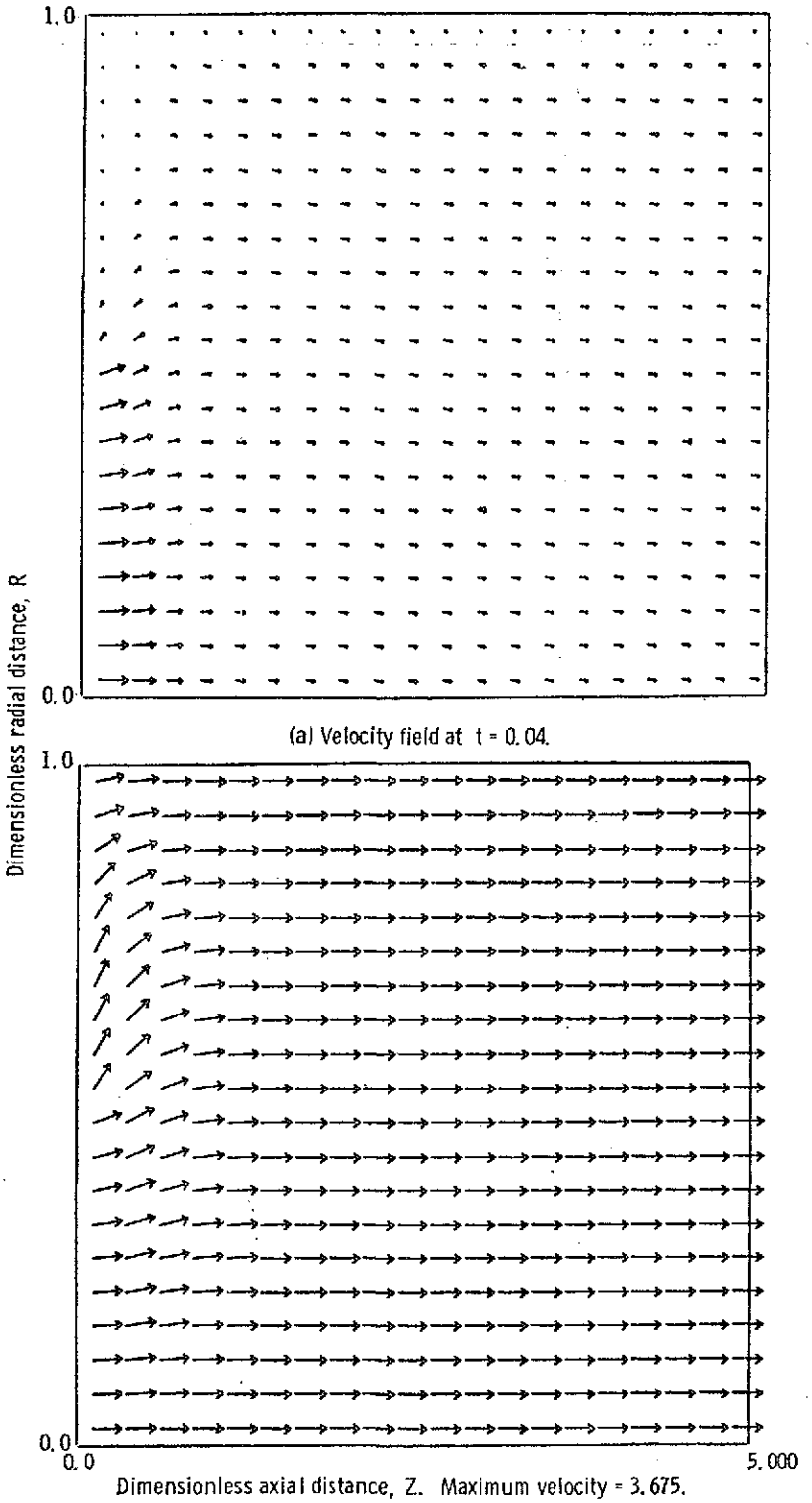
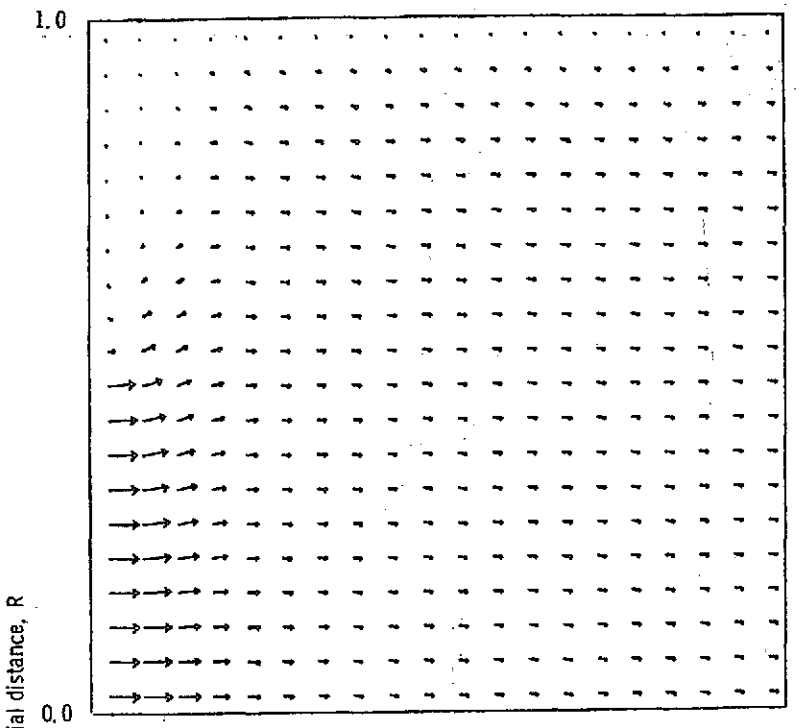
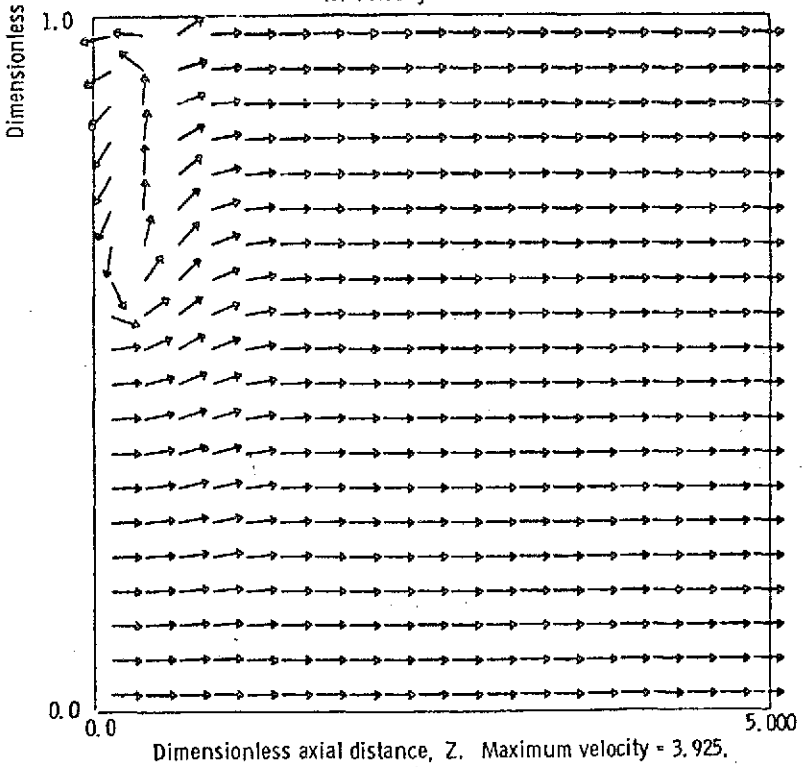


Figure 29. - Transient velocity vector and flow direction figures for coaxial entry, center jet only. Velocity = 20 cm/sec.

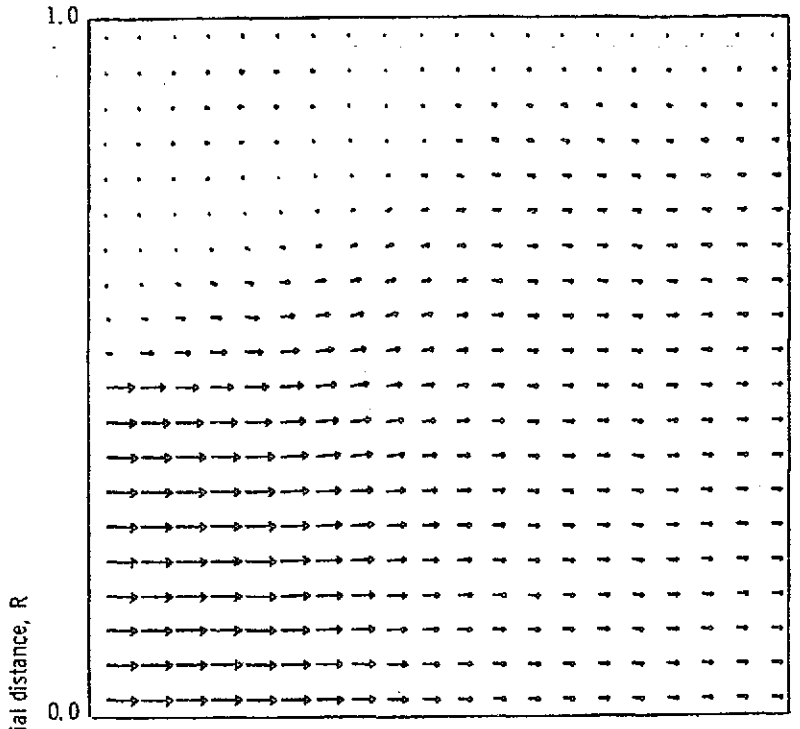


(c) Velocity field at t = 0.1.

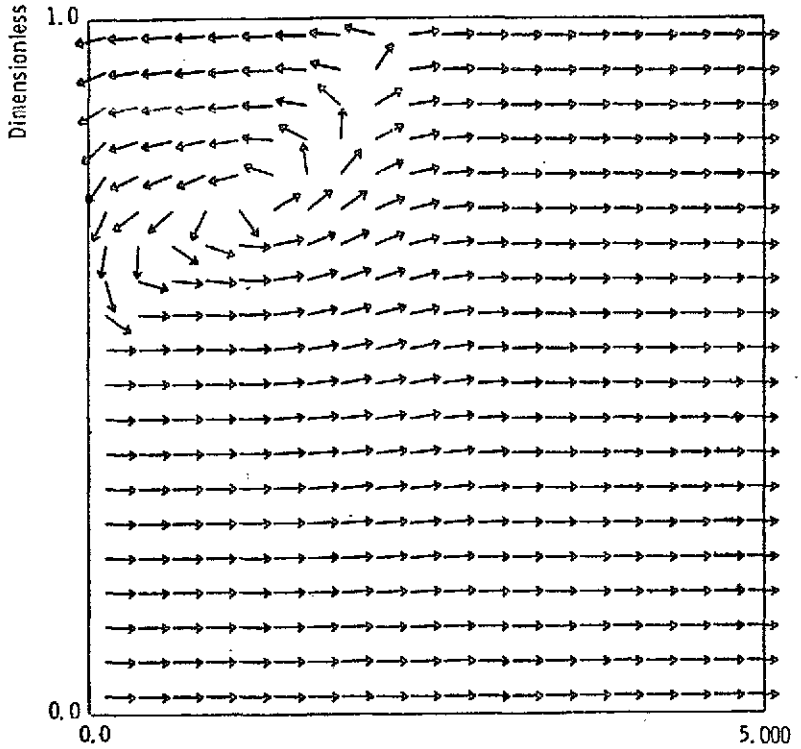


(d) Flow direction at t = 0.1.

Figure 29. - Continued.

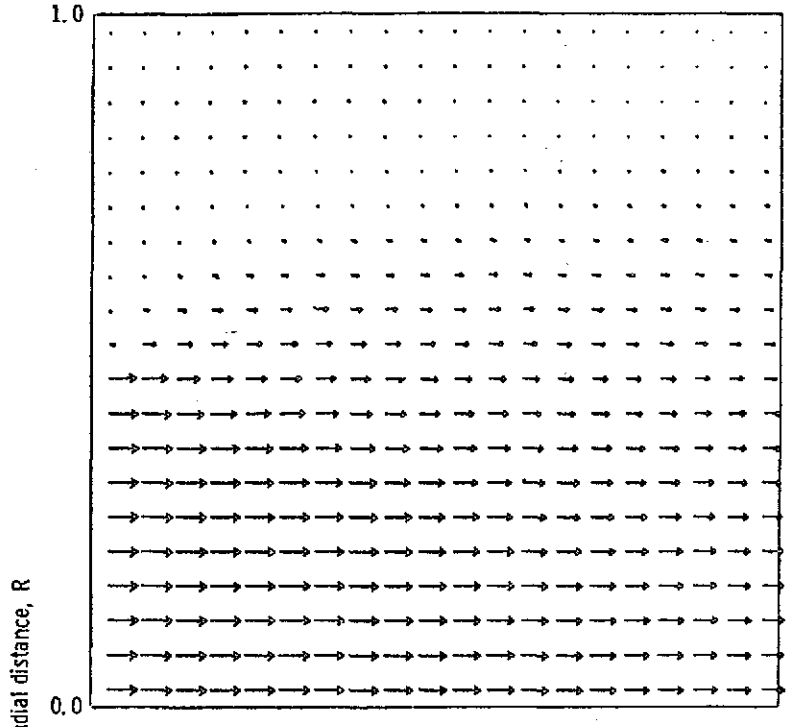


(e) Velocity field at t = 0.5.

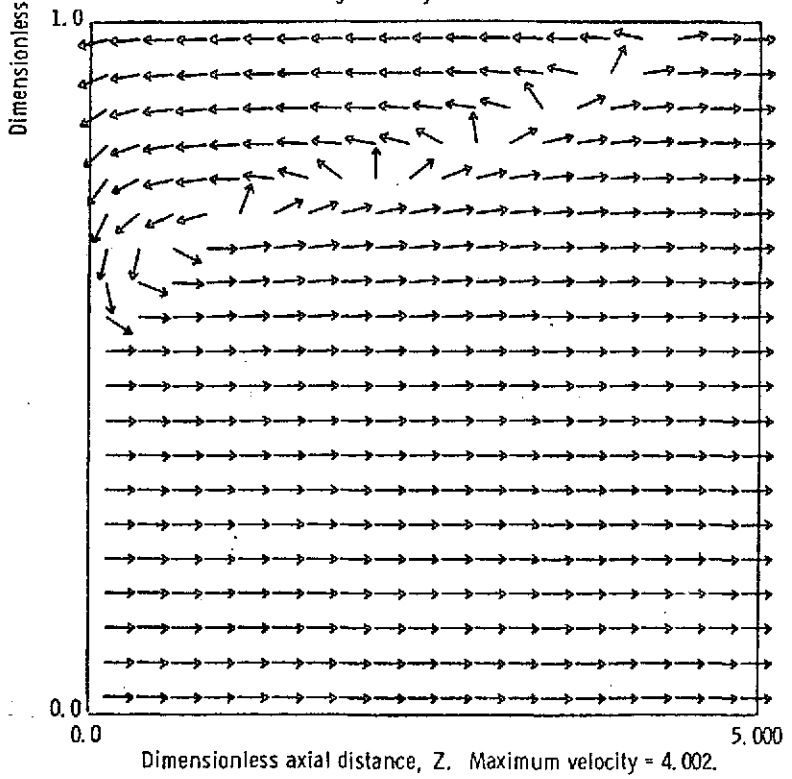


(f) Flow direction at t = 0.5.

Figure 29. - Continued.

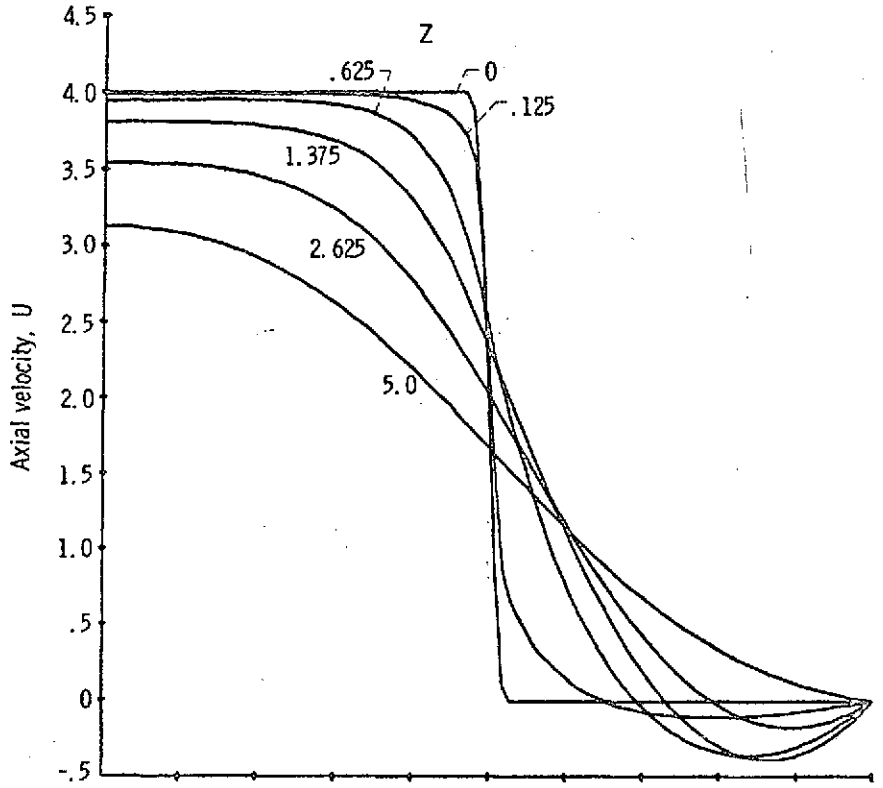


(g) Velocity field at t = 2.

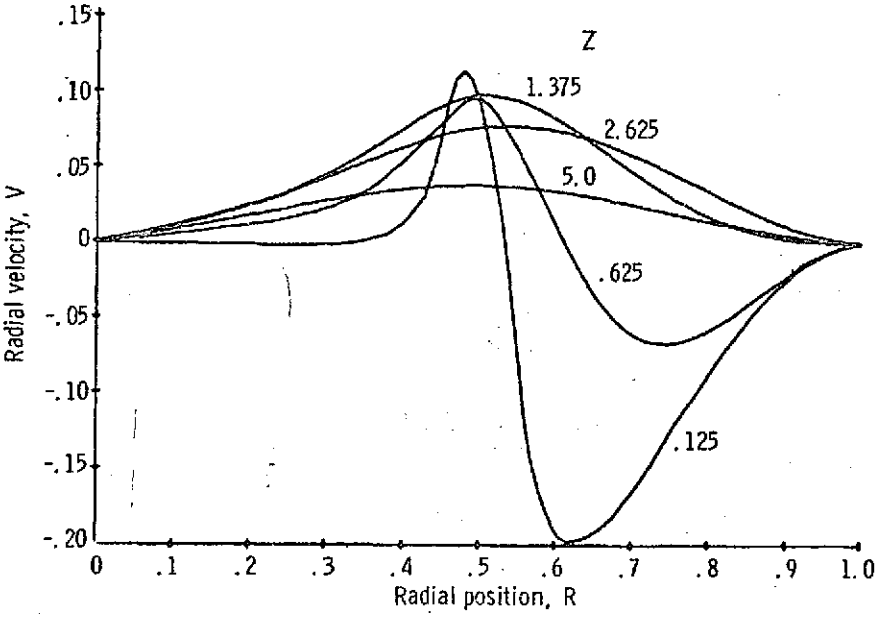


(h) Flow direction at t = 2.

Figure 29. - Concluded.

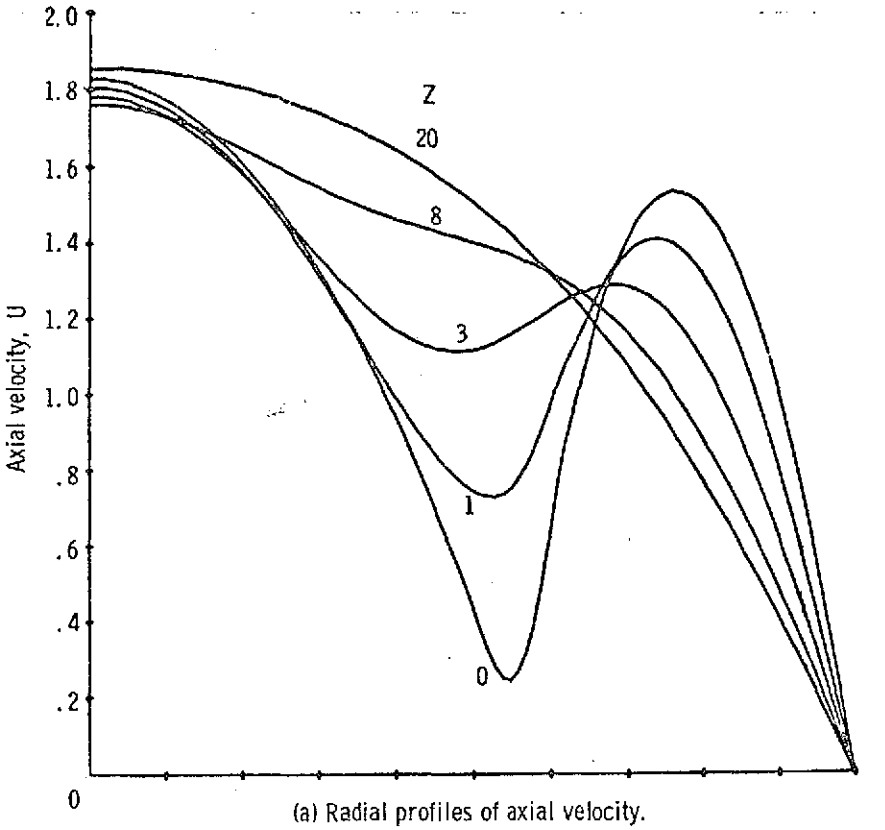


(a) Radial profiles of axial velocity.

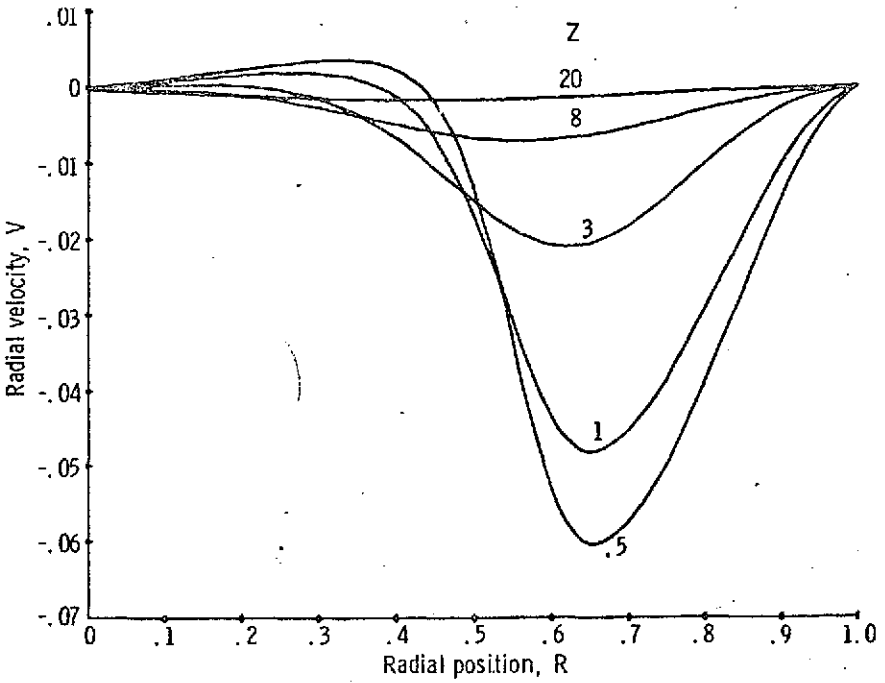


(b) Radial profiles of radial velocity.

Figure 30. - Coaxial entry, center jet only; t = 2.0 (steady state).



(a) Radial profiles of axial velocity.



(b) Radial profiles of radial velocity.

Figure 31. - Coaxial parabolic entry into a long tube with diffusion and reaction of trace species.

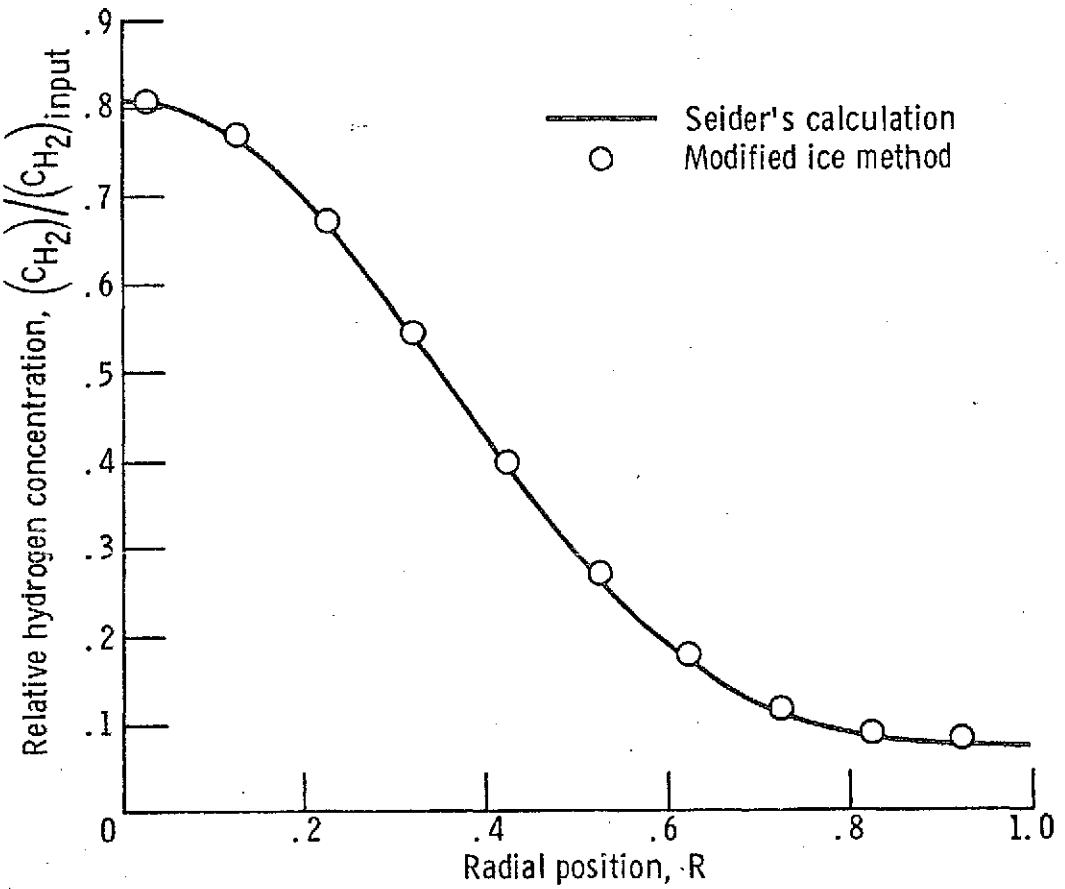
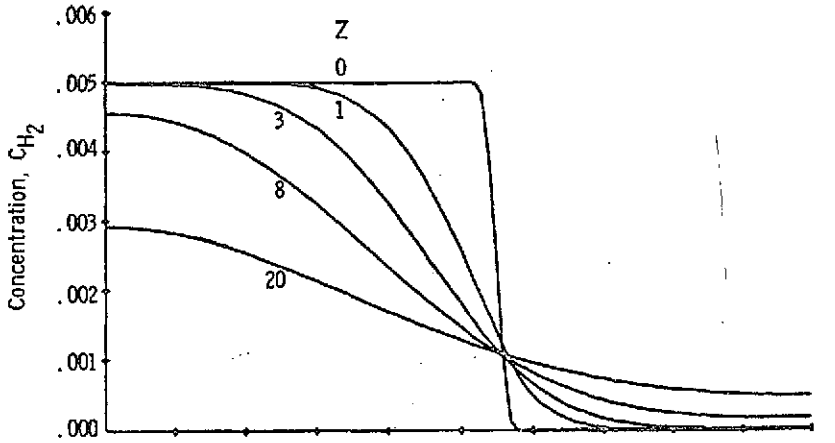
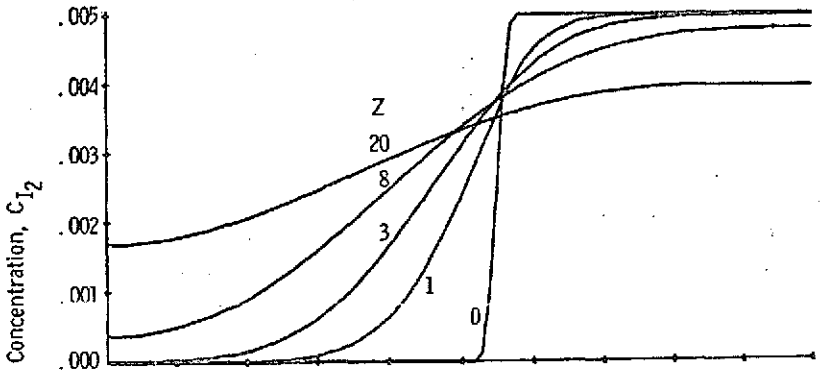


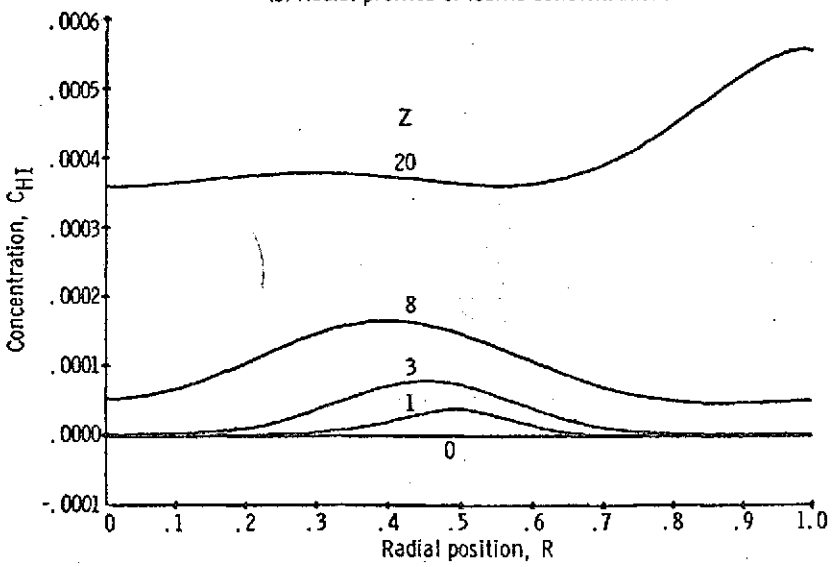
Figure 32. - Comparison with Seider's no-reaction computation $Z = 12$, $N_{Re} = 248$, $N_{Sc} = 0.942$.



(a) Radial profiles of hydrogen concentration.



(b) Radial profiles of iodine concentration.



(c) Radial profiles of hydrogen iodide concentration.

Figure 33. - Parabolic coaxial entry into a long tube with diffusion and reaction of trace species.

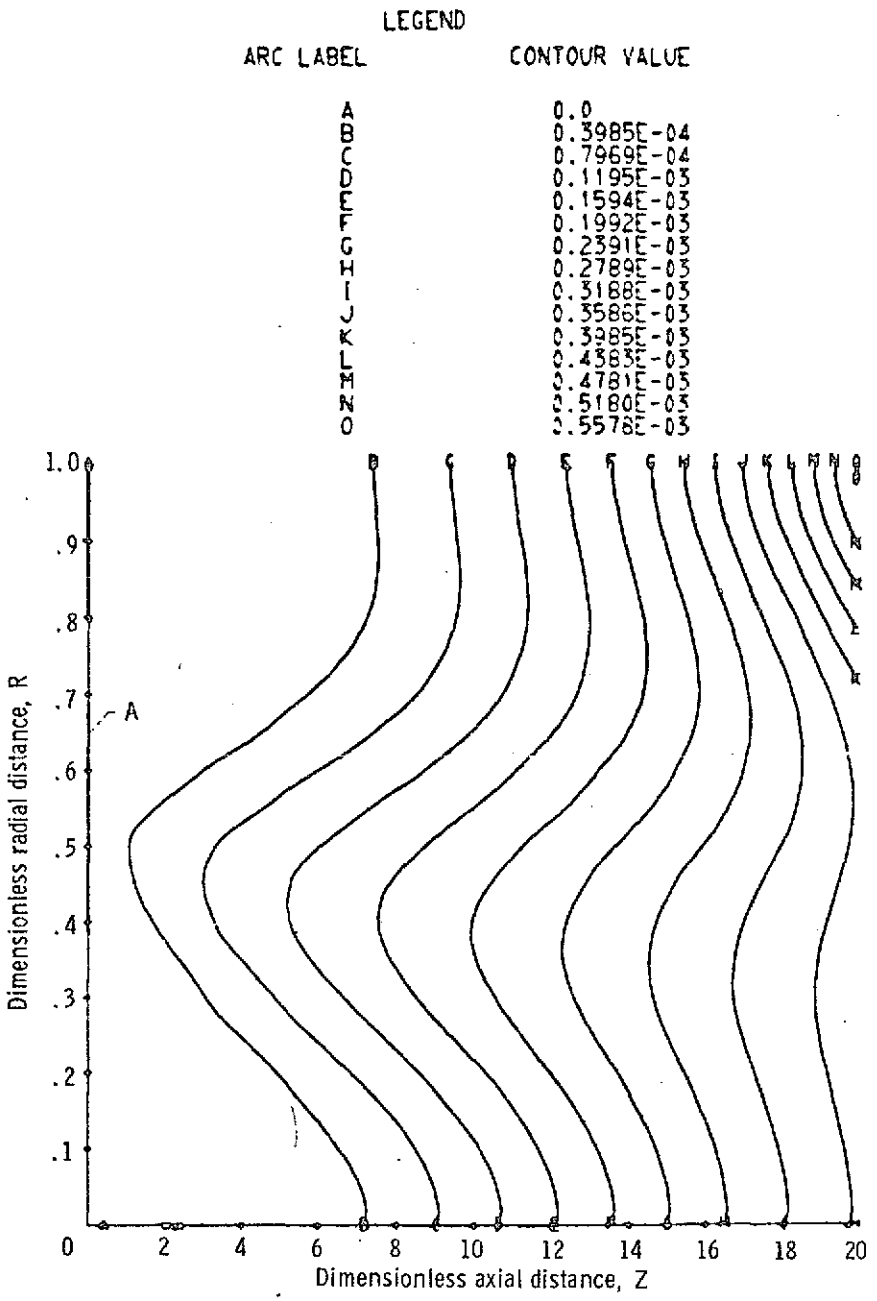
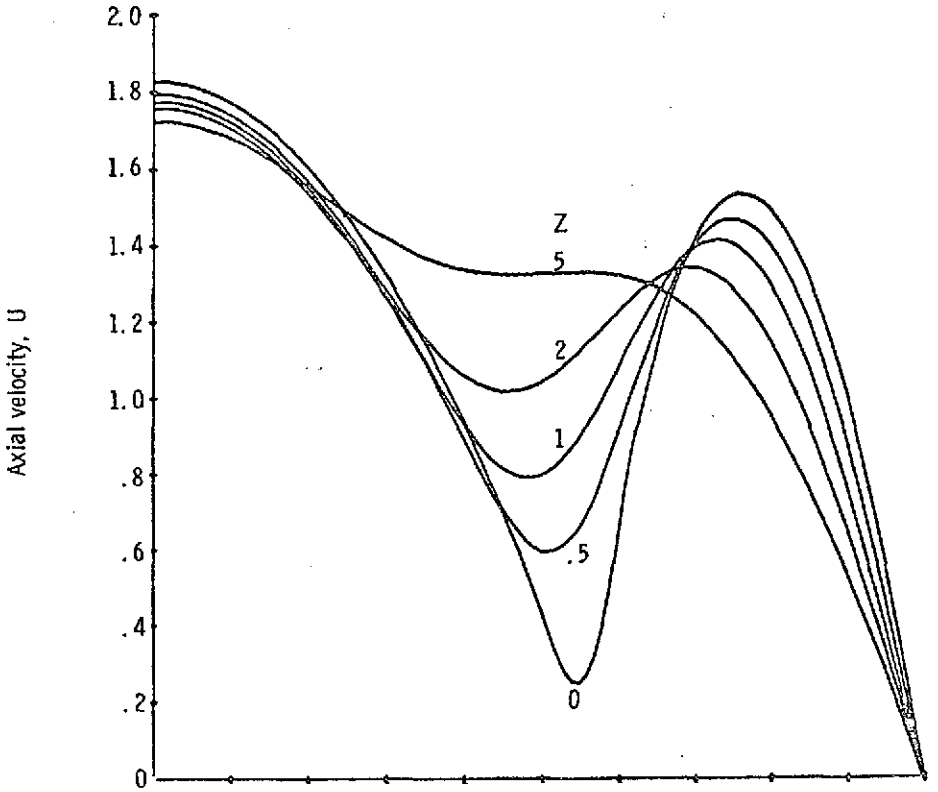
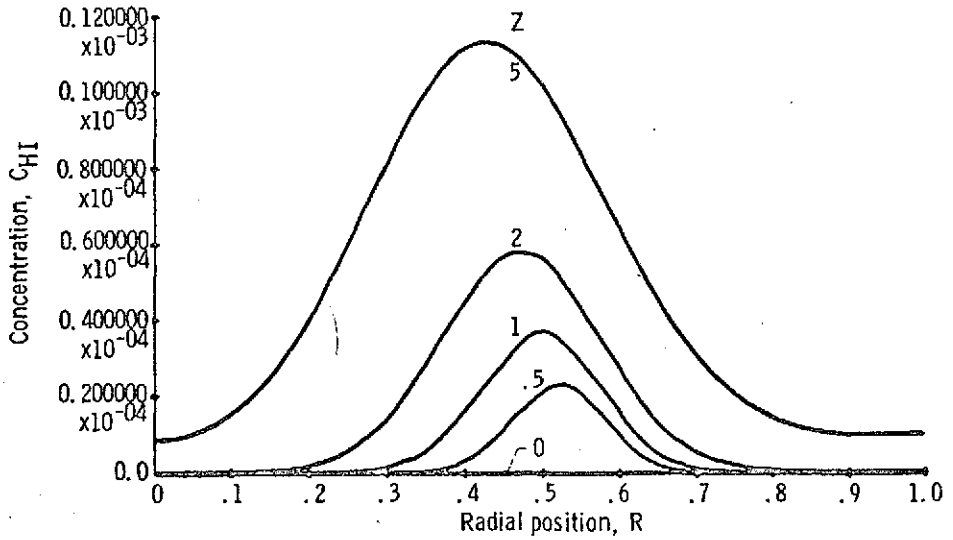


Figure 34. - Contour plot of hydrogen iodide concentration. Parabolic coaxial entry into a tube 20 radii long.



(a) Radial profiles of axial velocity.



(b) Radial profiles of hydrogen iodide concentration.

Figure 35. - Parabolic entry into a tube 5 radii long. Diffusion and reaction of trace species.

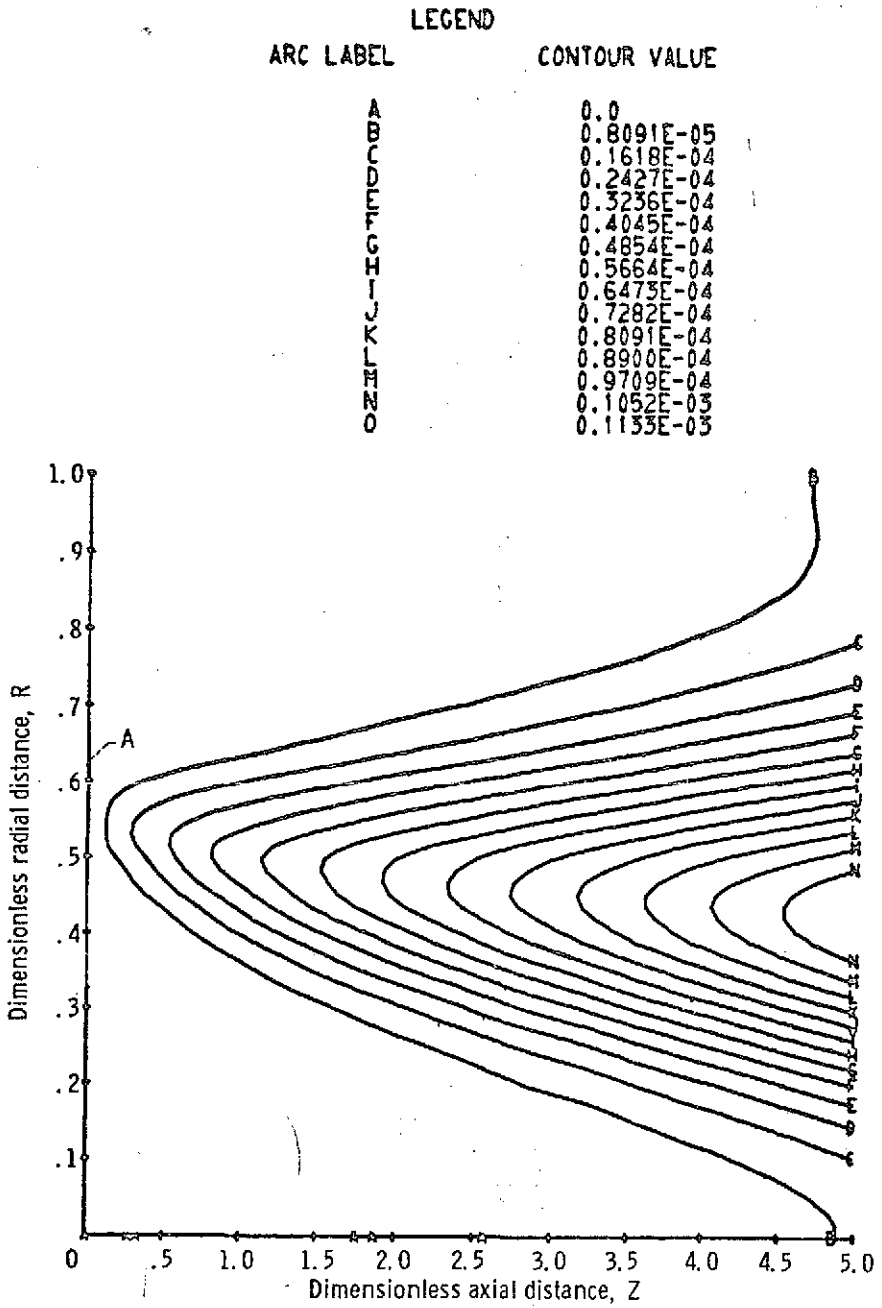


Figure 36. - Contour plot of hydrogen iodide concentration. Parabolic co-axial entry into a tube 5 radii long.

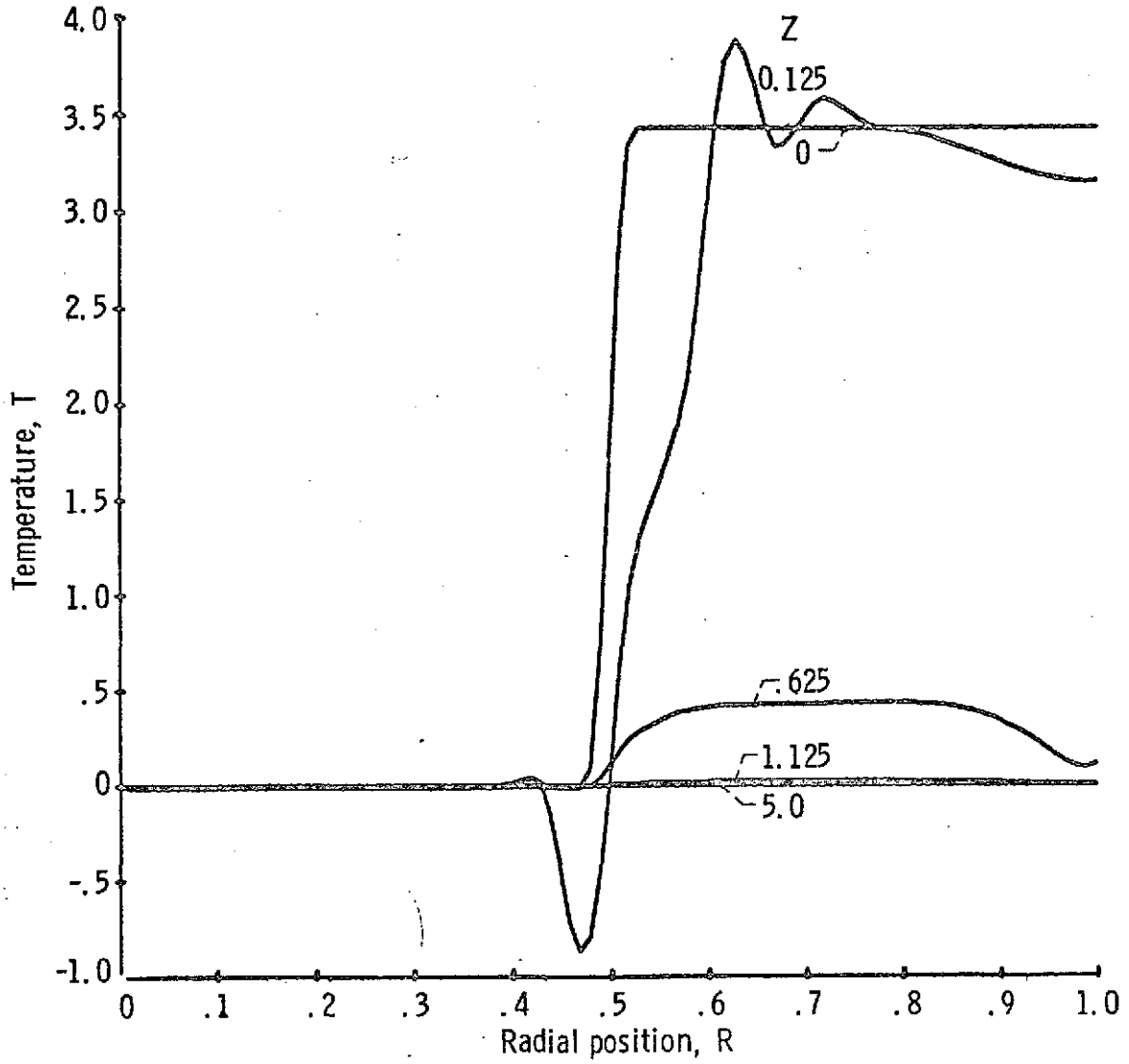


Figure 37. - Onset of instability in temperature profiles for the CO oxidation problem.

Appendix A

INVERSION OF THE $\tilde{P}$ MATRIX

The finite difference equation for the hybrid function $\tilde{P}$ is given by equation (4-26). This equation may be solved by several methods, one being simple relaxation. In this case the new values of $\tilde{P}_{i,j}$ are found by using equation (4-26) and sweeping through the grid, computing the $(Q + 1)$th value of $\tilde{P}$ from the Qth values surrounding it. After the entire net of $(Q + 1)$th values are computed they are substituted in the $\tilde{P}$ matrix and the process is repeated.

A variation is to control the convergence by multiplying the right hand side of (4-26) by a factor α while also adding a term which is $(1 - \alpha)$ times the old value of $\tilde{P}$. If $\alpha = 1$, the result is straight relaxation. If $0 < \alpha < 1$, the iteration is termed underrelaxation which converges more slowly and becomes more stable as α is made smaller. For overrelaxation, $1 < \alpha < 2$ and convergence is hastened up to a point, which for the problems run here was $\alpha = 1.8$. At higher values of α, convergence becomes more difficult or impossible. At $\alpha = 2$ the method is unstable.

Another method which usually accelerates convergence is the immediate replacement with the new value of P. This leads to the functional dependency shown in (4-27) if the grid is swept through increasing i and j. Successive overrelaxation (SOR) with immediate

substitution was used up to the tube entry flow problem. At that point cells with lower aspect ratio were used and convergence began to slow. This prompted examination and use of another method.

A modification of the alternating direction implicit method (ADI) by Brian (6) was used herein. It has been used by Schwab (46) among others and appears to offer some stability and convergence advantages plus higher order accuracy. Briefly, the time step is split in half and the finite difference equation is written as a three level scheme. The equation for the first half time step is written with all the differences in one coordinate in implicit form. For example, given the simple equation

$$\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial R^2} + \frac{\partial^2 F}{\partial Z^2} \tag{A-1}$$

the first half time step can be written

$$\frac{F^*_{i,j} - F^n_{i,j}}{\Delta t/2} = \frac{F^*_{i+1,j} - 2F^*_{i,j} + F^*_{i-1,j}}{(\Delta R)^2} + \frac{F^n_{i,j+1} - 2F^n_{i,j} + F^n_{i,j-1}}{(\Delta Z)^2} \tag{A-2}$$

This gives a tridiagonal set of equations which may be solved without iteration (see any numerical analysis text). Next the equation is written implicitly in the other coordinate, the values F^* being known.

$$\frac{F^{**}_{i,j} - F^n_{i,j}}{\Delta t/2} = \frac{F^*_{i+1,j} - 2F^*_{i,j} + F^*_{i-1,j}}{(\Delta R)^2} + \frac{F^{**}_{i,j+1} - 2F^{**}_{i,j} + F^{**}_{i,j-1}}{(\Delta Z)^2} \tag{A-3}$$

Again a tridiagonal system of equations is formed and solved for the $F^{**}_{i,j}$ values. Finally

$$\frac{F^{n+1}_{i,j} - F^{n}_{i,j}}{\Delta t} = \frac{F^{*}_{i+1,j} - 2F^{*}_{i,j} + F_{i-1,j}}{(\Delta R)^2} + \frac{F^{**}_{i,j+1} - 2F^{**}_{i,j} + F^{**}_{i,j-1}}{(\Delta Z)^2} \qquad \text{(A-4)}$$

which is explicit. In practice, the equations may be subtracted from each other to eliminate F^{**}.

The ADI method was used to solve the implicit species equation. It was also used to invert the $\tilde{P}$ matrix after converting that Poisson-type equation to a pseudo time dependent equation. This was done by clearing the denominator in equation (4-26), moving all terms to the right hand side, and replacing the zero on the left hand side with the finite difference analog to $\partial\tilde{P}/\partial X$, where "chi", X, is the pseudo-time. The ADI method was then applied and the equations solved as though the P variables were moving through the X domain to a steady state solution which was the set of values $\tilde{P}_{i,j}$ at $n + 1$. This was done using equations (4-28a) and (4-28b). In that sense, the solution of the $\tilde{P}$ equation was still iterative.

Tests with the entry problem showed that SOR converged the $\tilde{P}$ field faster than ADI if the cells were square. But the use of long rectangular cells, corresponding to low aspect ratio, gave ADI a speed advantage over SOR which was as high as a factor of five. Numerical experimentation showed that the convergence rate behaved as shown in figure 38. A few restarts varying ΔX and iterating over one Δt established the two slopes and the optimum value of ΔX. The ADI method was used for all problems with uniform and coaxial entry flow.

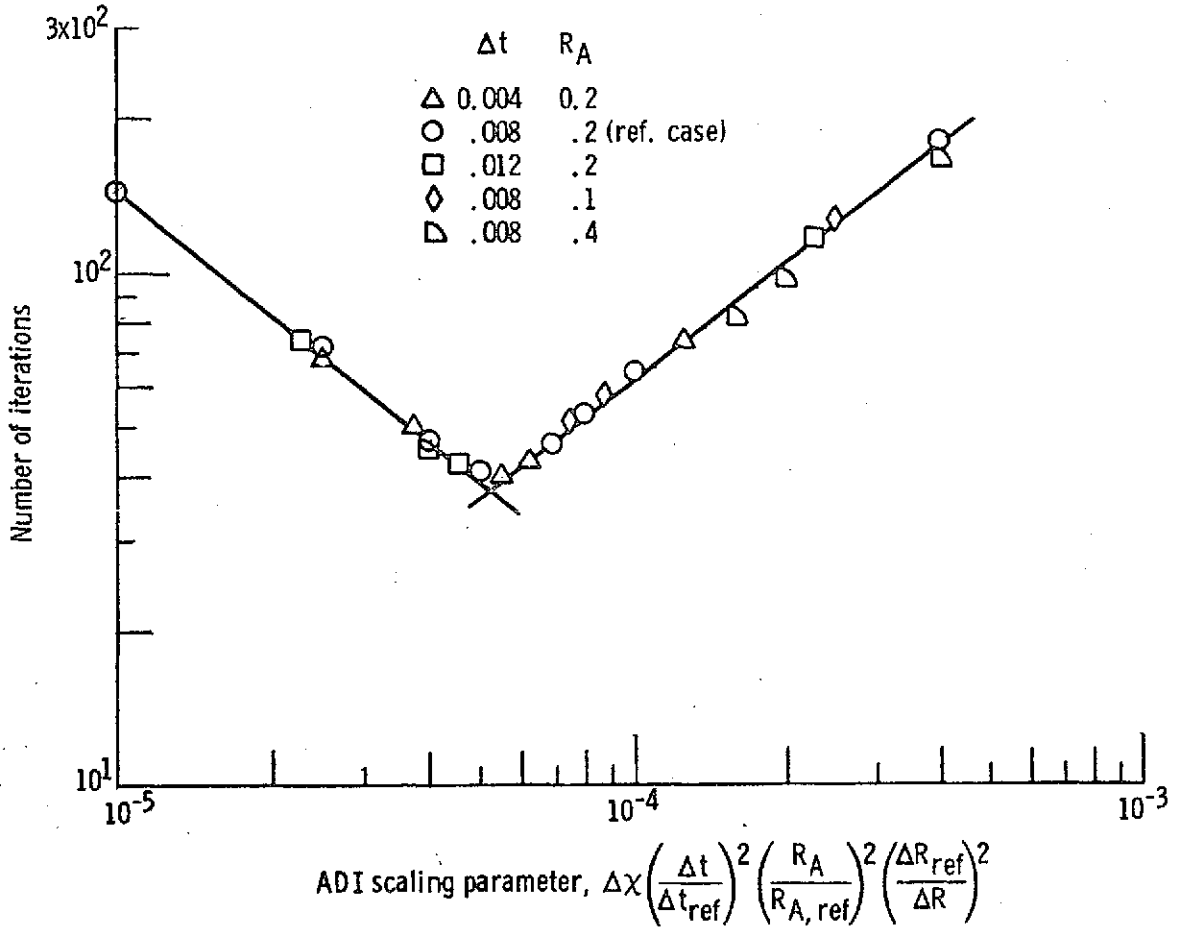


Figure 38. - Number of iterations required to converge the pressure field versus an ADI scaling parameter.

Appendix B

OUTLINE OF THE ALGORITHM FOR FLOWS WITH STRONG REACTION

Since the carbon monoxide problem was not completed, this appendix briefly discusses the planned approach for dealing with the kinetics and heat of reaction. The detailed kinetic equations for CO oxidation are quite extensive as shown in Brokaw and Bittker (8). A mass equation for each specie or radical would be needed, including data for the diffusion coefficients. To avoid this the computer program used in reference (8) was run for a series of concentrations and temperatures. The global rate equation for a second order reaction was integrated to give an equation predicting a dimensionless concentration as a function of a global rate and time. The results of the detailed computations were fit to this equation to get the global rate from the slope. Figure 39 is an example of such a plot. The good fit to a straight line indicates the global rate adequately describes CO oxidation for the temperature and initial concentration shown. Most fits were good, and a library of global rates was planned.

The intended sequence of computations of specie reaction was as follows:

1) Solve the species convection and diffusion over the time substep $\Delta t'$.

2) Compute the enthalpy of each cell based on the new composition and old temperature.

3) Enter the reaction computation, pick a global rate for each cell, and react at constant temperature over $\Delta t''$.

4) Compute enthalpy based on the new composition and the change that occured over $\Delta t''$.

5) Compute the heat capacities C_p and C_v and calculate the temperature change from the change in enthalpy. Calculate the energy change.

6) Repeat steps (3), (4), and (5) over all the $\Delta t''$ required until step (1) must again be repeated.

This sequence contains a number of errors. One is that for a mixture,

$$H = \sum_M H_M X_M \tag{B-1}$$

where H_M is a function of T for each specie M, the reaction causes both a change in composition and temperature. Thus

$$\Delta H = \sum (\Delta H_M X_M + H_M \Delta X_M) \tag{B-2}$$

and the second term has been ignored. The same problem occurs for internal energy. For small composition changes this error may be acceptable.

The choice of position in the algorithm for the species equation was arbitrary. But it was felt that computing mixing, reaction, and subsequent temperature changes just before calculating $P_{i,j}^{n+1}$ was the

most appropriate position. Thus the reaction affects the new pressure, and the pressure is the variable which couples the reaction chemistry and fluid flow. The pressure field immediately acts on the flow in the next time increment.

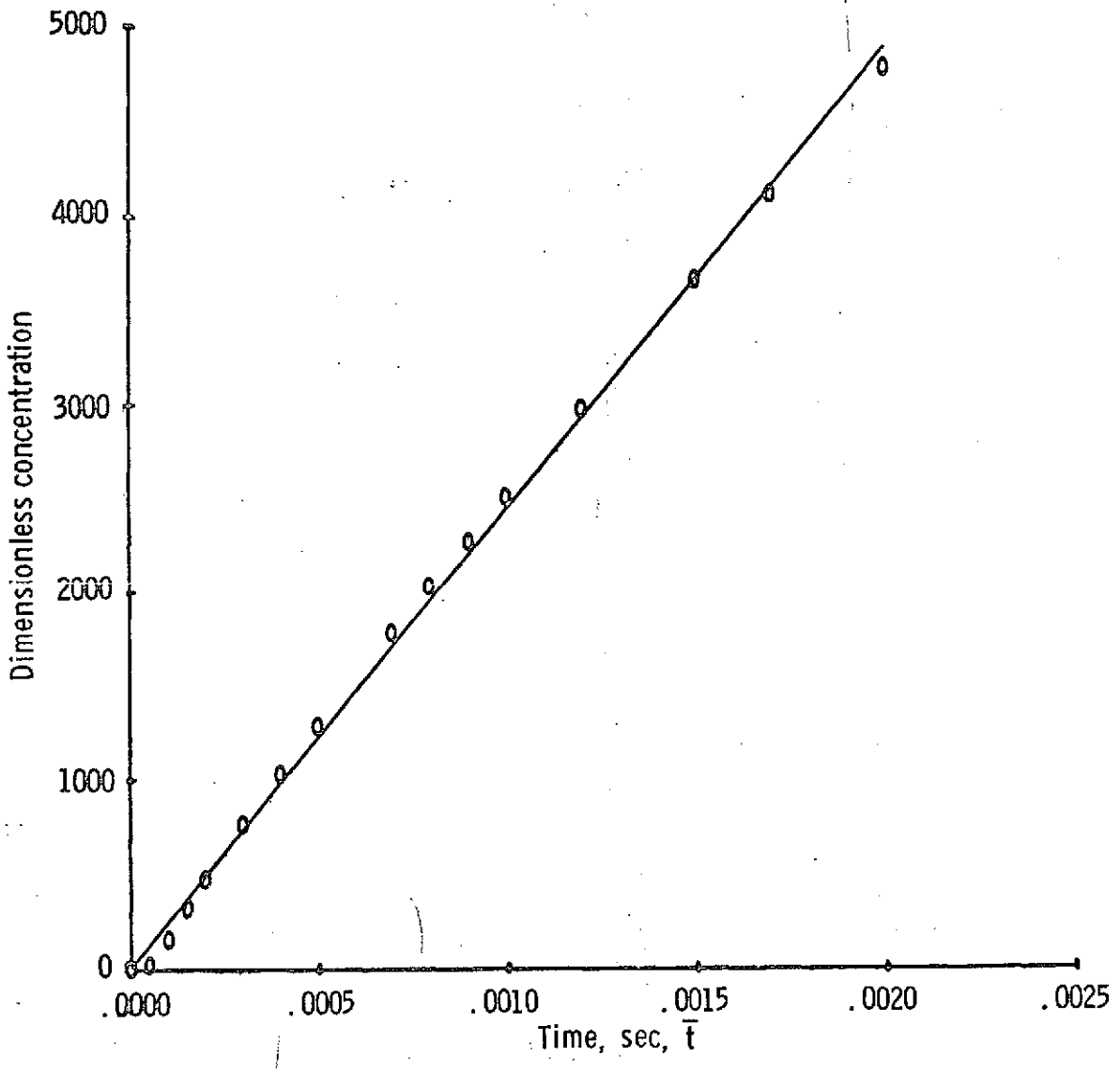


Figure 39. - Curve fit of detail kinetic calculations to give a global rate. Second order oxidation of carbon monoxide. $\bar{T}$ = 1600; initial mole fractions: CO, 0.10; O_2, 0.10; H_2O, 0.12; H_2, 0; CO_2, 0.10; N_2, 0.58; $CO + \frac{1}{2}O_2 = CO_2$; $dC_{CO}/dt = -k'' C_{CO} C_{O_2}^{1/2}$.

Appendix C

DESCRIPTION OF THE NUMERICAL PROGRAM

The program listed at the end of this appendix is an experimental program written for a time sharring computer. It is not in a "production" form, and it will require some effort to switch the program to batch mode operation. Much room for streamlining is certainly possible and desirable.

The program listing represents the status of the last problem attempted, that of carbon monoxide oxidation. The program is scaled for a maximum grid size of 20 x 40 and for five species. Lines of comment within the programs help explain the purpose of the program segments they block off.

The main program and subroutines will be identified. Next the important variable names will be defined. Many other variables are self evident and need not be defined. The equation symbols used in this thesis were chosen by convention (U = axial velocity) and because they could be identified with their variable (R = radius). The symbol system initially employed was more cumbersome and used many Greek symbols. Unfortunately the computer program uses variable words consistent with the old symbol system. Some common ones are:

UR = radial velocity, V

VZ = axial velocity, U

SQUIG = radius, R

TAU = time, t

RDL = aspect ratio, R_A

SIG = density, ρ

A. Programs

FMIX6 - Main program which controls all flow of information, and performs major calculations.

INIT - Subroutine which impresses the initial conditions of the problem, plus sets some constants.

BOUN - Subroutine which provides boundary conditions.

PROPS6 - Subroutine which specifies the properties of the fluids and computed the reaction.

PADI - Subroutine which performs the ADI iteration of the $\tilde{P}$ equation.

MASS - Subroutine which solves the species equation.

RESET - Subroutine which permits resetting the variables in the TSS mode when all other programs are compiled with the Internal Symbol Dictionary default, ISD = n. Can be excised in a batch program.

RITE - Subroutine which writes output and tapes plotting data.

SIMPLE - Subroutine which calculated reaction fluxes by Simpsons Rule.

B. Equation Variables, Constants and Supporting Information

A1→F1, B2, E2	Coefficients for the specie ADI.
B	Explicit term in radial momentum equation.
BC1→BC8	Constants in boundary conditions.
BETAK	Input for β_{MK} coefficient.

BETAM	Input for β_M coefficient.
BETAMT	Mass truncation error correction.
BETAV	Input for β_{VR} and β_{VZ} coefficients.
BETAVR	Radial momentum truncation error correction.
BETAVZ	Axial momentum truncation error correction.
BKCOEF	Coefficient for β_{MK} correction.
BMCOEF	Coefficient for β_M correction.
BVCOEF	Coefficient for β_{VR} and β_{VZ} corrections.
C1IN	Input specie concentrations in center tube of coax.
→C5IN	
C1OUT	Input specie concentrations in annulus of coax.
→C5OUT	
C1STOR	Arrays for tape storage
→C5STOR	
CC1	Main equation coefficients.
→CC42	
CODV	Curve fit coefficients for diffusion coefficients.
COK	Curve fit coefficients for thermal conductivity.
COMU	Curve fit coefficients for viscosity.
CONC	Total concentration.
CONC1	Concentration of the 5 species.
→CONC5	
CONCUP	Upstream total concentration.
CONUP1	Upstream species concentrations.
→CONUP5	
CONOLD	Old value of continuity.
CONT	New value of continuity.
COTHER	Coefficients for heat capacity.
CPIN	Center tube input fluid heat capacity.
CPN2	Nitrogen C_P.
CPO2	Oxygen C_P.
CPOW	Annulus input fluid heat capacity at constant pressure.
CPSTAR	Reference heat capacity at constant pressure.
CSTAR	Reference concentration.
CVN2	Nitrogen C_V.
CVO2	Oxygen C_V.
CVSTAR	Reference C_V.
D	Explicit terms in axial momentum equation.
DCHI	Pseudo-time step ΔX.
DI, EI, FI, DJ, EJ, FJ	ADI coefficients for $\tilde{P}$ and ρ_K.
DSQIG	Radial cell dimension, ΔR.
DTAU	Time increment, Δt.
DV	Binary-type diffusion coefficient,
DVSTAR	Reference diffusion coefficient,
ENEW	n + 1 value of total energy, E.
ENG	n value of total energy, E.
ENGN	Normalized value of total energy.
EOLD	Old value of energy.

EULNO	Euler number, N_{Eu}.
EY	Normalized reference internal energy.
EYE	Internal energy, I.
EYEN	Normalized internal energy.
EYEOW	Internal energy of annulus input flow.
EYOLD	Old value of internal energy.
EYSTAR	Reference internal energy, I^*.
FREQ	Cyclic frequency, f.
G	Explicit term in $\tilde{P}$ equation.
GAMIN	Center tube input fluid.
GAMMY	Ratio of heat capacities,
GAMOW	Annulus input fluid.
I	Radial cell index.
J	Axial cell index.
LT	Total length.
MFINIT	Input mole fraction.
MLES	Mixture Molecular weight minus specie K.
MLESIN	MLES for the coaxial center tube input.
MLESOW	MLES for the coaxial annulus input.
MMIXIN	Mixture molecular weight for the coaxial center tube input.
MMIXOW	Mixture molecular weight for the annulus input.
MOLFR	Mole fraction, X.
MOLMIX	Mixture molecular weight.
MOLWT	Normalized molecular weight, M.
MSTAR	Reference molecular weight, M^*.
MUSTAR	Reference velocosity, μ^*.
MWT	Specie molecular weight.
NPR	Prandtl number, N_{Pr}.
NRE	Reynolds number, N_{Re}.
NSC	Schmidt number, N_{Sc}.
PB	Pressure, $\tilde{P}$.
PBHAF	Value of $\tilde{P}$ after first half of ADI iteration.
PHI	Coefficient,
PR	Pressure, P.
PRANIN	Prandtl number for center tube coax input.
PRANNO	Prandtl number.
PRANOW	Prandtl number for annulus coax input.
PRSTOR	Pressure to be taped.
PSI	Coefficient, ψ.
PSTAR	Reference pressure, P^*.
PUP	Upstream pressure.
Q	Velocity divergence.
RAT1	Radius ratio (i - 1)/(2i - 3).
RAT2	Radius ratio (i - 2)/(2i - 3).
RAT3	Radius ratio (2i - 5)/(i - 2).
RDL	Aspect ratio, R_A.
REYNIN	Reynolds number for center tube input.
REYNO	Reynolds number.
REYNOW	Reynolds number for annulus input.
RIN	Radius of coax center tube.
ROUT	Radius of coax outer tube, R_W.

S	Explicit terms in the species equation.
SGSTAR	Reference density, ρ^*.
SGSTOR	Density to be taped.
SIG	Density, ρ.
SIGK	Specie density, ρ_K.
SIGOW	Density for annulus input.
SKHAF	Value of specie density after first half of ADI iteration.
SKOLD	Old value of specie density.
SMITIN	Schmidt number for coax center tube input.
SMITNO	Schmidt number, N_{Sc}.
SMITOW	Schmidt number for coax annulus input.
SOLD	Old density.
SOUN	Speed of sound squared, Á.
STRONO	Strouhal number, $N_{S\ell}$.
T	Temperature, T.
TAU	Time, t.
TAUEND	Time at the end of a run.
TCYC	Cycle time, the reference time
TEMPIN	Temperature of coax center tube input.
TEMPOW	Temperature of coax annulus input.
THETA	Coefficient in mass equation.
TIME	Time value for taping.
TPSTOR	Temperature value for taping.
TSTAR	Reference temperature, T^*.
UR	Radial velocity, V.
UROLD	Old radial velocity.
URSTOR	Radial velocity for taping.
VIN	Velocity for coax center tube input.
VOUT	Velocity for coax annulus input.
VREF	Reference velocity.
VUP	Upstream velocity.
VZ	Axial velocity, U.
VZOLD	Old axial velocity.
VZSTOR	Axial velocity for taping.

C. Program Control Variables

CELLIN	The number of the last cell within the center tube for annular flows.
CONTHI	Largest acceptable value of the error on the continuity equation. If exceeded, the time step is cut.
CONTLO	Low value of continuity error. If the error is less than this number for 3 consecutive times, the time step increases.
DTMAX	Maximum allowable time step.
ERROR	$\tilde{P}$ convergence error (also ERROR1).
ILES	NR + 2.
IMAX	NR + 3.
IMIN	NR + 1.
IMNI	NR

INCOMP	If value = 1, the fluid is incompressible.
ITERS	The number of $\tilde{P}$ iterations in that time step.
ITEST	The radial cell number denoting the $\tilde{P}$ value being tested for convergence.
ITMAX	The maximum allowable $\tilde{P}$ iterations in one time cycle
JEND	Largest axial index for computing U.
JLES	NL + 2.
JMAX	NL + 3.
JMIN	NL + 1.
JMNI	NL.
JSTART	Smallest axial index for computing U.
JTEST	The axial cell number denoting the $\tilde{P}$ value being tested for convergence.
NANN	The lowest cell number within the annulus.
NBUG	If value is 1, debug output is written.
NCONT	Counter on continuity test for increasing Δt.
NCOUNT	Loop counter on writeout test.
NFIRST	If value = 0, the $\tilde{P}$ array calculated in the first Δt is saved.
NFIRPB	If the value is 1, the previously stored $\tilde{P}$ array is loaded at initialization of the problem.
NFLU	Identifies the fluid being used.
NIN	Number of $\Delta t'$ in each Δt for the solution of the species equation.
NITER	The $\tilde{P}$ field iteration number.
NL	Number of within-grid axial cells.
NOUT	Number of allowable outer iterations on I^{n+1}.
NOUTER	Outer iteration number.
NPRITE	Counter on $\tilde{P}$ iteration for writeout of $\tilde{P}$ array.
NPROB	The problem number.
NR	Number of within-grid radial cells.
NREACT	If value = 1, reaction is calculated.
NRITE	If value = 1, printer output is used.
NRUN	Run number.
NSAVE	Sets the number of Δt increments between renewing restart data storage.
NSPECY	Number of species in multicomponent calculations.
NSTOP	If value = 1, program stops.
NSTART	If value = 1, a restart is effected by reading dataset with last restart storage.
NT	Number of time loops.
NTAPE	If value = 1, output is stored on tape for later plotting.
NTLOOP	Time loop counter for output.
NTRITE	Test value for writeout.
NTUBE	Largest cell number within the center tube of coaxial injection.
NY	Number of $\Delta t''$ within each $\Delta t'$ for computing reaction.

D. Listing of Computer Program

```
      COMMON/AA/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/BB/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAI,BETAK,DCHI
      COMMON/CC/BKCOEF,CC15,CC17,CC32,CC33,CC34,CC35,CC36,CC37
      COMMON/RR/NTRITE,ITMAX,NPRITE,NFLU,NPROP,NR,NL,NT,NITER,NOUT
      COMMON/SS/TAUEND,NRITE,NTAPE,NSAVE,NFIRPP
      COMMON/TT/DOLD,BETAMT,BETAVR,BETAV7,JSTART
      COMMON/WW/RAT1,RAT2
      COMMON/XX/RIN,VIN,VOLT,NTUBE,NANN
      COMMON/YY/DTOLD,NDTAU,NINCLD,NYOLD
      COMMON/AB/MSTAR,TSTAR,CSTAR,DVSTAR,CC38,SPECIF
      COMMON/AC/NRE,NPR,GAMMY,MOLMIX,CONC
      COMMON/AD/NSC
      COMMON/AE/CC2,CC8,CC9,CC16,NPROP,NSPECY,NSPEC,MULTID,NY,NIN,NREACT
      COMMON/AF/SIGK,MOLER
      COMMON/AH/SIGKUP,SKUP
      COMMON/AK/CC26,CC27
      COMMON/AL/MLES,T,GAMIN,GAMOW,TEMPOW,SIZEX
      COMMON/AM/MOLFUP,MMIXUP,MLESUP,SMITUP,GAMUP,SMITIN,SMITOW,-
Z     REYNIN,REYNOW,REYUP
      COMMON/AN/MMIXIN,MMIXOW,MLESIN,MLESOW
      COMMON/AO/EYEOW,ENGOW,SIGOW,TPOW
C
    1 FORMAT(T2,'CPUTIME ='F9.3,' SECONDS.')
    2 FORMAT(1H4,T40,50HXXXXX  X       XXX   X   X  X   X  X  X   X  XXXXX/T40,-
Z                    50HX     X      X   X  X   X  XX XX  X   X X   X    /T40,-
Z                    50HXXX   X      X   X  X X X  X X X  X    X    XXXXX/T40,-
Z                    50HX     X      X   X  XX XX  X   X  X   X X   X   X/T40,-
Z                    50HX     XXXXX   XXX   X   X  X   X  X  X   X  XXXXX//)
    9 FORMAT(T2,'READ SIZE: SIZER=DTAU,TCYC,RCLT,RDL,PHI,THETA,PSI,-
ZBETAM,BETAV,BETAK,TAUEND,DCHI,ERROR'/,T13,'SIZEI=ITMAX,NTRITE,-
ZNPRITE,NR,NL,NPROB,NFLU,NREACT,NY,NIN,NOUT,NFIRPP,NRITE,NTAPE,NRUN')
    4 FORMAT(T2,'DTAU='F10.8,' TCYC='F6.4,' RCLT='F6.3,' RDL='F4.2,-
Z     ' PHI='F4.2,' THETA='F4.2,' PSI='F4.2,' BETAM='F4.2,' BETAV='-
Z     F4.2,' BETAK='F4.2/,T2,'TAUEND='F8.4,' DCHI='F9.8,' ERROR='-
Z     1PE7.1)
    5 FORMAT(T2,'ITMAX='I5,' NTRITE='I3,' NPRITE='I5,' NP='I2,' NL='I2,-
Z     ' NPROB='I2,' NFLU='I1,' NREACT='I1,' NY='I2,' NIN='I2,' NOUT='I2/,-
Z     T2,'NFIRPB='I1,' NRITE='I1,' NTAPE='I1,' NRUN='I3)
    6 FORMAT(T2,I4,' ITERATIONS DONE. I QUIT.')
    7 FORMAT(1H1)
    8 FORMAT(1H ,25I5)
    9 FORMAT(T2,'IF THIS IS A RESTART, TYPE T')
   10 FORMAT(T2,'READ IN COMMENTS ON THIS RUN, FORMAT 120H0----.')
   11 FORMAT(120H0                                                          -
Z                                                                          /)
   12 FORMAT(L1)
   13 FORMAT(T2,'ERROR='1PE10.4)
C
C     -----START MAIN PROGRAM WITH TIME CHECK-----
C
   50 CALL CPUTIM (MLSEC)
      SECONE=MLSEC
C     -----READ INPUT AND INITIALIZE PROBLEM-----
  100 NRUG=0
```

```
C FMIX6 - WIEBER PROGRAM FOR UNSTEADY, TWO-DIMENSIONAL, MULTICOMPONENT
C         TUBE FLOWS. USES THE ICE METHOD FOR ALL SPEED FLOWS -
C         INCLUDES VARIABLE OUTER ITERATION CAPABILITY, AUTOMATIC ADJUSTMENT
C         OF ACCEPTABLE ERROR IN THE PRESSURE FIELD ITERATION, MANUAL
C         ADJUSTMENT OF TIME AND WRITEOUT STEPSIZE, AND PERFORMS THE
C         PRESSURE FIELD ITERATION BY THE ALTERNATING DIRECTION IMPLICIT
C         TECHNIQUE USING SUBROUTINE PADI# - MASS AND MOMENTUM CORRECTIONS IN
C         REQUIRES SUBROUTINES INIT#, BOUN#, PROP#, RITE#, RESET, MASS2, PADI#
C         & PADI#. INCLUDES MIXING AND REACTION OF UP TO 5 SPECIES.
C
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION PR(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
     Z EYE(23,43),ENG(23,43)
      DIMENSION QOLD(23,43),G(23,43)
      DIMENSION P(23,43),D(23,43),SOLD(23,43)
      DIMENSION PB(23,43),CONOLD(23,43)
      DIMENSION SOUN(23,43),CONT(23,43)
      DIMENSION Q(23,43),ENEW(23,43)
      DIMENSION RAT1(23),RAT2(23),RAT3(23)
      DIMENSION BETAMT(23,43),BETAVR(23,43),BETAVZ(23,43)
      DIMENSION GOLD(23,43),EYOLD(23,43),-
     Z VZOLD(23,43),UROLD(23,43),EOLD(23,43)
      DIMENSION AI(23),CI(23),DI(23),EI(23),FI(23),DJ(43),EJ(43),FJ(43)
      DIMENSION GAMUP(23),CONC(23,43),GAMMY(23,43),T(23,43)
      DIMENSION SIGK(23,43,5),SIGKUP(23,5),SKUP(23,5)
      DIMENSION SIZEX(6),SMITUP(23,4),REYUP(23),SMITOW(4),SMITIN(4)
C
      REAL*8 L,LT,MOLWT(5),MLSTAR,LDR,LDRSQ,SIZER(13)
      REAL*8 MOLMIX(23,43),NPR(23,43),NRE(23,43)
      REAL*8 MSTAR,MOLFR(23,43,5),MSC(23,43,4)
      REAL*8 MMIXIN,MMIXOW,MLESIN(4),MLESOW(4),MLES(23,43,4)
      REAL*8 MOLFUP(23,5),MMIXUP(23),MLESUP(23,4)
C
      INTEGER DAY,ITERS(4000),SIZEI(15),SPECIE(6)
C
      LOGICAL ANS
C
      NAMELIST/SIZE/SIZER,SIZEI
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/BB/EYOLD,VZOLD,UROLD,EOLD
      COMMON/CC/PR,SIG,UR,VZ,EYE,ENG
      COMMON/DD/BC4,BC6,ERRS,ITEST,JTEST,NSTOP,NCOUNT,ERROR1
      COMMON/EE/PB,PTIME,TAU
      COMMON/FF/GOLD,G
      COMMON/GG/PUP,PUPMAX,SIGUP,SUPMAX,DSCIG,VELUP,VUPMAX,RDL,FREQ
      COMMON/HH/R,D,SOLD,SUP2,SUP2
      COMMON/II/NRUG,NRUN,MONTH,DAY
      COMMON/JJ/AI,BI1,BI2,CI,DI,EI,FI,AJ,BJ1,BJ2,CJ,DJ,EJ,FJ,CC41,CC42
      COMMON/KK/SOUN,CONT
      COMMON/LL/EYSTAR,ROUT,LT,EULNO,REYNO,STRONO,SMITNO,PRANNO,VREF
      COMMON/MM/Q,ENEW
```

```
      NSTART=0
      NFIRPB=0
      ANS=.FALSE.
      WRITE (69,9)
      READ (69,12) ANS
      IF (.NOT.ANS) GO TO 103
      NSTART=1
  101 IF (NSTART.EQ.0) GO TO 103
  102 READ (10) PB,PR,V7,UR,SCUN,ENG,EYE,SIG,DTAU,TCYC,ROUT,RCL,PHI,-
     Z THETA,PSI,BETAM,BETAV,BETAI,BETAK,T,DCHI,ERROR,TAU,TAUEND,VUP,VREF,-
     Z PUP,PUP2,SIGUP,EY,VIN,VCUT,CELLIN,RIN,FRACP,JSTART,ITMAX,NTRITE,-
     Z NPRITE,NR,NL,NPROB,NFLU,IACCMP,NCUT,NRUN,NT,NNN,NTUBE,NANN,NSAVE,-
     Z NRITE,NTAPE,NTLOOP,NCOUNT,NREACT,NY,NIN,NSPECY,(ITERS(I),I=1,NNN)
      PAUSE 'RESTART, CHANGES ON DTAU, DCHI, OR NSAVE?'
      GO TO 110
  103 WRITE (69,3)
      READ (5,SIZE)
  104 WRITE (69,4) SIZER
      WRITE (69,5) SIZEI
      PAUSE 'ALL OKAY?'
  105 DTAU=SIZER(1)
      TCYC=SIZER(2)
      RCUT=SIZER(3)
      RCL=SIZER(4)
      PHI=SIZER(5)
      THETA=SIZER(6)
      PSI=SIZER(7)
      BETAM=SIZER(8)
      BETAV=SIZER(9)
      BETAK=SIZER(10)
      TAUEND=SIZER(11)
      DCHI=SIZER(12)
      ERROR=SIZER(13)
  106 ITMAX=SIZEI(1)
      NTRITE=SIZEI(2)
      NPRITE=SIZEI(3)
      NR=SIZEI(4)
      NL=SIZEI(5)
      NPROB=SIZEI(6)
      NFLU=SIZEI(7)
      NREACT=SIZEI(8)
      NY=SIZEI(9)
      NIN=SIZEI(10)
      NCUT=SIZEI(11)
      NFIRPB=SIZEI(12)
      NRITE=SIZEI(13)
      NTAPE=SIZEI(14)
      NRUN=SIZEI(15)
C     -----SET WRITING INDICES-----
  110 IMAX=NR+3
      ILES=IMAX-1
      IMIN=IMAX-2
      IMAI=NR
      JMAX=NL+3
```

```
      JLES=JMAX-1
      JMIN=JMAX-2
      JMAI=NL
C     -----SET MISC. CONSTANTS-----
  120 DSCIG=1./AR
      DSCISQ=DSCIG*DSCIG
  121 RDLSQ=RDL*RDL
      FREQ=1./TCYC
      CONTHI=2.5D-3
      CONTLO=5.0D-4
  124 IF (NCUT.EQ.0) NCUT=1
      BMCOEF=0.
      IF (BETAM.GT.1.) BMCOEF=BETAM-1.
      BVCOEF=0.
      IF (BETAV.GT.1.) BVCOEF=BETAV-1.
      BKCOEF=0.
      IF (BETAK.GT.1.) BKCOEF=BETAK-1.
      IF (NSTART.EQ.1) GO TO 126
  125 NT=0
      NNN=0
      NFIRST=1
      NTLOOP=0
      NCOUNT=0
      NSAVE=1
      NSPECY=1
      INCOMP=0
      TAL=0.
      NRESET=0
 1250 DO 1251 I=1,4000
      IF (ITERS(I).EQ.0) GO TO 126
      ITERS(I)=0
 1251 CONTINUE
  126 NITER=0
      NSAME=0
      NPROP=0
  127 LT=DSQIG*ROUT/RDL*NL
C     -----GET FLUID PROPERTIES-----
  130 CALL PROP
      IF (NSTART.EQ.0) GO TO 140
      IF (NSPECY.EQ.1) GO TO 132
      READ (10) ARE,NPR,GAMMY,MOLMIX,CONC,SIGK,MOLPR,MLES,GAMIN,-
     Z  GAMOW,TEMPOW,SIZEX,MMIXIN,MMIXOW,MLESIN,MLESOW,MOLFUP,MMIXUP,-
     Z  MLESUP,SMITUP,GAMUP,SMITOW,REYNIN,REYNOW,REYUP,SIGKUP,SKUP,-
     Z  NSC,EYECK,ENGOW,SIGCK,TPCK
  132 REWIND 10
      GO TO 141
C     -----INITIALIZE------
  140 CALL INIT
  141 CTMAX=.9*SQSTAR*RCUT*RCUT*DSQISQ/(2.*MUSTAR*TCYC*(1.+RDLSQ))
C     -----CALCULATE DIMENSIONLESS GROUPS AND CC# COEFFICIENTS-----
  155 REYNO=SQSTAR*ROUT*VREF/MUSTAR
      EULNO=PSTAR/(SQSTAR*VREF*VREF)
      STRONO=ROUT/(VREF*TCYC)
      SMITNO=MUSTAR/(SQSTAR*DVSTAR)
```

```
1550 IF (NSTART.EQ.1) GO TO 156
     IF (INCOMP.EQ.1) GO TO 156
     IF (PHI.GT.0.99.AND.THETA.GT.0.99) GO TO 156
1551 PHI=1.
     THETA=1.
     NRESET=1
 156 CC1=8./3.
     CC2=.5*RDL
     CC3=RDL/16.
     CC4=2./DSQIG
     CC5=2.*RDL
     CC6=(2.*THETA*DTAU*DTAU)/(DSQISQ*STRONC*STRONO)
     CC7=EULNO*(1.-PHI)
     CC8=.5*RDLSQ
     CC9=DTAU/(DSQIG*STRONC)
     CC10=.25*RDL*CC9
 157 CC11=2.*CC9
     CC12=EULNO*PHI
     CC13=RDL*CC12
     CC14=RDL*CC7
     CC15=.25/DSQIG
     CC16=CC15*RDL
     CC17=DSQIG/PSI
     CC18=STRONC/DTAU
     CC19=THETA/DSQIG
 158 CC20=(1.-THETA)/DSQIG
     CC21=.5*CC9
     CC22=EULNC*CC9
     CC23=CC21/DSQIG
     CC24=CC23*CC2
     CC25=4.*RDL
     CC26=CC9/(8.*DSQIG*NIN)
     CC27=CC26*RDLSQ
     CC28=.75*CC9
     CC29=CC9/64.
     CC30=CC28*RDL
     CC31=CC29*RDL
     CC32=2.*NIN/(PSI*CC9)
     CC33=CC32/CC2
     CC34=CC33*CC33
     CC35=(1.-PSI)/DSQIG
     CC36=4./(DSQIG*DSQIG)
     CC37=CC36*CC8
     CC38=SQSTAR*TCYC*DTAU/(NY*NIN)
 159 CC40=CC6*CC12
     CC41=2./DCHI
     CC42=.5*CC40*RDLSQ
     BI1=-CC40-CC41
     BI2=BI1-4.*CC41
     AJ=CC42
     BJ1=-2.*CC42-CC41
     BJ2=BJ1-4.*CC41
     CJ=CC42
     PHI=SIZER(5)
```

```
      THETA=SIZER(6)
C     -----CALCULATE RADIUS COEFFICIENTS-----
  160 DO 164 I=2,ILES
  161 XI=I
      XIXI=XI*XI
  162 RAT1(I)=(XI-1.)/(XIXI-2.)
      RAT2(I)=(XI-2.)/(XIXI-3.)
      AI(I)=CC40*RAT2(I)
      CI(I)=CC4C*RAT1(I)
      IF (I.EC.2) GO TO 164
  163 RAT3(I)=(XIXI-5.)/(XI-2.)
  164 CONTINUE
C     -----INITIALIZE------
  170 CALL RCUN
C     -----READ IN COMMENTS-----
  180 WRITE (69,10)
      READ (69,11)
C     -----READ IN INITIAL PB FIELD IF DESIRED-----
  185 IF (NSTART.EQ.1) GO TO 190
      IF (NFIRPB.EC.0) GC TC 190
  186 READ (14) PB
      REWIND 14
C     -----SET OLD VALUES = INITIAL VALUES-----
  190 DO 196 J=1,JMAX
      VZOLD(1,J)=VZ(1,J)
      URCLD(IMAX,J)=UR(IMAX,J)
  191 DO 195 I=2,ILES
  192 VZOLD(I,J)=VZ(I,J)
      URCLD(I,J)=UR(I,J)
  193 ECLD(I,J)=ENG(I,J)
      EYCLD(I,J)=EYE(I,J)
  194 SOLD(I,J)=SIG(I,J)
  195 CONTINUE
  196 CONTINUE
      IF (INCOMP.EC.1) JEND=JMIN
      IF (INCOMP.EQ.0) JEND=JLES
  197 IF (NSTART.EQ.0) GC TC 198
      NSTART=0
      IF (NTAPE.EC.0) GO TO 200
      ENDFILE 12
      BACKSPACE 12
      GC TC 200
  198 WRITE (6,2)
      WRITE (6,11)
  199 CALL RITE
C
C     -----ENTER MAIN TIME LOOP-----
C
  200 TAU=TAU+DTAU
      NTLOOP=NTLOOP+1
      NCOUNT=NCOUNT+1
      NCUTER=C
      IF (NPROP.EQ.1) GC TC 201
      CALL PROPC
```

```
      IF (NSPECY.GT.1) CALL BCUNO
      CALL BCUNF
  201 IF (NITER.EQ.0) GO TO 510
C----------CALCULATE TOTAL MASS ERROR CORRECTIONS-----
  202 IF (BETAM.EQ.0.) GO TO 210
  203 DO 209 J=2,JMIN
      JP=J+1
  204 DO 208 I=2,IMIN
      IP=I+1
      DIFFU=CC21*(UR(IP,J)-UR(I,J))
      DIFFV=CC10*(VZ(I,JP)-VZ(I,J))
  205 IF (BETAM.GT.1.) GO TO 206
      DIFFU=BETAM*DIFFU
      DIFFV=BETAM*DIFFV
      GO TO 207
  206 DIFFU=BMCOEF*CABS(DIFFU)
      DIFFV=BMCOEF*CABS(DIFFV)
  207 BETAMT(I,J)=DIFFU*(RAT1(I)*SIG(IP,J)+RAT2(I)*SIG(I-1,J)-
     Z    -SIG(I,J))+DIFFV*(SIG(I,JP)+SIG(I,J-1)-2.*SIG(I,J))
  208 CONTINUE
  209 CONTINUE
C     -----CALCULATE OLD PART OF CONTINUITY-----
  210 DO 215 J=2,JMIN
      JP=J+1
      JM=J-1
  211 DO 214 I=2,IMIN
      IP=I+1
      IM=I-1
  212 S=SOLD(I,J)
      SIP=SOLD(IP,J)
      SIM=SOLD(IM,J)
      SJP=SOLD(I,JP)
      SJM=SOLD(I,JM)
  213 CONOLD(I,J)=-CC18*S+CC20*(RAT1(I)*(SIP+S)*UROLD(IP,J)-RAT2(I)*(S-
     Z    +SIM)*UROLD(I,J)+CC2*((SJP+S)*VZOLD(I,JP)-(S+SJM)*VZOLD(I,J)))-
     Z    -CC18*BETAMT(I,J)
  214 CONTINUE
  215 CONTINUE
C     -----CALCULATE B & D-----
  220 DO 230 J=2,JLES
      JP=J+1
      JM=J-1
  221 DO 229 I=2,ILES
      IP=I+1
      IM=I-1
      R1=RAT1(I)
      R2=RAT2(I)
C 222 U=UR(I,J)
      UIP=UR(IP,J)
C     UIM=UR(IM,J)
C     UJP=UR(I,JP)
C     UJM=UR(I,JM)
C     UIPJM=UR(IP,JM)
C 223 V=VZ(I,J)
```

```
C       VIP=VZ(IP,J)
C       VIM=VZ(IM,J)
C       VJP=VZ(I,JP)
C       VJM=VZ(I,JM)
C       VIMJP=VZ(IM,JP)
C  224  S=SIG(I,J)
C       SIP=SIG(IP,J)
C       SIM=SIG(IM,J)
C       SJP=SIG(I,JP)
C       SJM=SIG(I,JM)
C       SIPJM=SIG(IP,JM)
C       SIMJP=SIG(IM,JP)
C       SIMJM=SIG(IM,JM)
    225 IF (BETAV.EQ.0.) GO TO 226
        IF (I.EQ.2) GO TO 2250
        XI2=I-2
        VSUM=VJP+V+VIMJP+VIM
        SI=S+SIM
        DIFFUR=(CC28*SI*U+.1875*SI/XI2+.125*(S-SIM))*U
        DIFFUZ=(CC31*SI*VSUM*VSUM+.015625*VSUM*(SJP+SIMJP-SJM-SIMJM)-
     Z  +.125*SI*(VJP+VIMJP-V-VIM))*RDL
     2250 XI3=I+I-3
        USUM=UIP+U+UIPJM+UJM
        SJ=S+SJM
        DIFFVR=CC29*SJ*USUM*USUM+.015625*USUM*(SIP+SIPJM-SIM-SIMJM)-
     Z  +.0625*SJ*(UIP+UIPJM-U-UJM)+.0625*SJ*USUM/XI3
        DIFFVZ=(CC30*SJ*V+.125*(S-SJM))*RDL*V
     2251 IF (BETAV.GT.1.) GO TO 2252
        DIFFUR=BETAV*DIFFUR
        DIFFUZ=BETAV*DIFFUZ
        DIFFVR=BETAV*DIFFVR
        DIFFVZ=BETAV*DIFFVZ
        GO TO 2260
     2252 DIFFUR=BVCOEF*DABS(DIFFUR)
        DIFFUZ=BVCOEF*DABS(DIFFUZ)
        DIFFVR=BVCOEF*DABS(DIFFVR)
        DIFFVZ=BVCOEF*DABS(DIFFVZ)
     2260 BETAVR(I,J)=DIFFUR*(UIP+UIM-U-U)+DIFFUZ*(UJP+UJM-U-U)
        BETAVZ(I,J)=DIFFVR*(VIP+VIM-V-V)+DIFFVZ*(VJP+VJM-V-V)
    226 PROD=(S+SJM+SIM+SIMJM)*(U+UJM)*(V+VIM)
        IF (I.EQ.2) GO TO 228
    227 B(I,J)=.5*U*(RAT3(I)*SIM*UIM-S*UIP/R2)+CC3*(PROD-(SJP+SIMJP+S-
     Z  +SIM)*(UJP+UI*(VJP+VIMJP))+CC4*((Q(I,J)-C(IM,J))+RDLSQ*(UJP+UJM-
     Z  -U-U)-RDL*(VJP-VIMJP-V+VIM))/(NRE(I,J)+NRE(IM,J))+BETAVR(I,J)
    228 D(I,J)=.125*(R2*PROD-R1*(SIP+S+SIPJM+SJM)*(UIP+UIPJM)*(VIP+V))-
     Z  +RDL*V*(SJM*VJM-S*VJP)+CC4*(RDL*(Q(I,J)-C(I,JM))+CC5*(R2*(U-UJM)-
     Z  -R1*(UIP-UIPJM))+2.*(R1*VIP+R2*VIM-V))/(NRE(I,J)+NRE(I,JM))-
     Z  +BETAVZ(I,J)
    229 CONTINUE
    230 CONTINUE
    231 B(IMIN,JLES)=0.
        B(ILES,JLES)=0.
        D(ILES,JMIN)=0.
        D(ILES,JLES)=0.
```

```
      CALL PROPA
  232 IF (NOUTER.NE.0) GO TO 270
C     -----CALCULATE OLD G-----
  240 DO 260 J=2,JMIN
      JP=J+1
      JM=J-1
  241 DO 259 I=2,IMIN
      IP=I+1
      IM=I-1
  242 IF (INCOMP.EQ.1) GO TO 255
  244 SOUN(I,J)=GAMMY(I,J)*EYOLD(I,J)/EULNO
  255 R1=RAT1(I)
      R2=RAT2(I)
  256 P=PR(I,J)
      S=SIG(I,J)
      SIP=SIG(IP,J)
      SIM=SIG(IM,J)
      SJP=SIG(I,JP)
      SJM=SIG(I,JM)
  257 GOLD(I,J)=P/SOUN(I,J)+CC6*(CC7*(R1*PR(IP,J)+R2*PR(IM,J)-P+CC8-
     Z    *(PR(I,JP)+PR(I,JM)-P-P))+R2*B(I,J)-R1*B(IP,J)+CC2*(D(I,J)-
     Z    -D(I,JP)))+CC9*(R2*(S+SIM)*UR(I,J)-R1*(SIP+S)*UR(IP,J)+CC2-
     Z    *((S+SJM)*VZ(I,J)-(SJP+S)*VZ(I,JP)))+BETANT(I,J)
  258 G(I,J)=GOLD(I,J)
  259 CONTINUE
  260 CONTINUE
      IF (INCOMP.EQ.1) GO TO 263
  261 DO 262 I=2,IMIN
      SOUN(I,1)=GAMMY(I,1)*EYOLD(I,1)/EULNO
      SOUN(I,JLES)=GAMMY(I,JLES)*EYOLD(I,JLES)/EULNO
  262 CONTINUE
  263 IF (NOUT.EQ.1) GO TO 300
C     -----CALCULATE TOTAL G-----
  270 DO 274 J=2,JMIN
  271 DO 273 I=2,IMIN
  272 G(I,J)=GOLD(I,J)-SOLD(I,J)*(1.-EYE(I,J)/EYOLD(I,J))
  273 CONTINUE
  274 CONTINUE
C     -----PRESSURE ITERATION-----
  300 CALL PADJ
  301 IF (NSTOP.EQ.0) GO TO 390
  302 IF (NFIRST.EQ.0) GO TO 370
      WRITE (14) PB
      REWIND 14
      NFIRST=0
C     -----WRITEOUT & STOP FOR NO PRESSURE CONVERGENCE-----
  370 WRITE (6,6) NITER
  371 NBUG=1
  377 CALL RITEB
      NNN=NNN+1
      ITERS(NNN)=NITER
  378 WRITE (6,7)
      WRITE (6,8) (ITERS(K),K=1,NNN)
  379 CALL CPLTIM (MLSEC)
```

```
      SECTHC=MLSEC
      SEC=(SECTHO-SECONE)/1000.
      WRITE (69,1) SEC
      STOP
  390 NNN=NNN+1
      ITERS(NNN)=NITER
C     -----CALCULATE DENSITY-----
  400 DO 404 J=2,JMIN
  401 DO 403 I=2,IMIN
  402 SIG(I,J)=(PB(I,J)-PR(I,J))/SCON(I,J)+SCLD(I,J)*(2.-EYE(I,J)-
     Z    /EYOLD(I,J))
  403 CONTINUE
  404 CONTINUE
  410 CALL BCUNP
C     -----CALCULATE VELOCITIES-----
  450 DO 456 J=2,JEND
      JM=J-1
  451 DO 455 I=2,IMIN
      IM=I-1
      IF (J.LT.JSTART) GO TO 453
  452 VZ(I,J)=((SCLD(I,J)+SCLD(I,JM))*VZOLD(I,J)+CC11*(CC13*(PB(I,JM)-
     Z    -PB(I,J))+CC14*(PR(I,JM)-PR(I,J))+B(I,J)))/(SIG(I,J)+SIG(I,JM))
  453 IF (I.EQ.2) GO TO 455
      IF (J.EQ.JLES) GO TO 455
  454 UR(I,J)=((SOLD(I,J)+SCLD(IM,J))*UROLD(I,J)+CC1)*(CC12*(PB(IM,J)-
     Z    -PB(I,J))+CC7*(PR(IM,J)-PR(I,J))+B(I,J)))/(SIG(I,J)+SIG(IM,J))
  455 CONTINUE
  456 CONTINUE
  460 CALL BCUNP
C     -----STORE OLD Q-----
  500 IF (NBUG.EQ.0) GO TO 510
      IF (NOUTER.NE.0) GO TO 510
      IF (MOD(NCOUNT,NTRITE).NE.0) GO TO 510
  501 DO 505 J=1,JLES
  502 DO 504 I=2,ILES
  503 QOLD(I,J)=Q(I,J)
  504 CONTINUE
  505 CONTINUE
C     -----CALCULATE NEW Q-----
  510 DO 514 J=1,JLES
      JP=J+1
  511 DO 513 I=2,ILES
  512 Q(I,J)=CC1*((RAT1(I)*UR(I+1,J)-RAT2(I)*UR(I,J))+CC2*(VZ(I,JP)-
     Z    -VZ(I,J)))
  513 CONTINUE
  514 CONTINUE
      Q(ILES,1)=0.
      Q(ILES,JLES)=0.
  515 IF (NITER.EQ.0) GO TO 202
C     -----CALCULATE CONTINUITY INCLUDING THE NEW PART-----
  550 DO 554 J=2,JMIN
      JP=J+1
      JM=J-1
  551 DO 553 I=2,IMIN
```

```
      IP=I+1
      S=SIG(I,J)
  552 CONT(I,J)=CONOLD(I,J)+CC18*S+CC19*(RAT1(I)*(SIG(IP,J)+S)*UR(IP,J)-
     Z    -RAT2(I)*(S+SIG(I-1,J))*UR(I,J)+CC2*((SIG(I,JP)+S)*VZ(I,JP)-(S-
     Z    +SIG(I,JM))*VZ(I,J)))
  553 CONTINUE
  554 CONTINUE
C     -----TEST CONT FOR ERROR ADJUSTMENT-----
  560 NCONT=0
  561 DO 566 J=2,JMIN
  562 DO 565 I=2,IMIN
  563 CABS=CABS(CONT(I,J))
      IF (CABS.GT.CONTHI) GO TO 567
      IF (CABS.GT.CONTLO) NCONT=1
  565 CONTINUE
  566 CONTINUE
      GO TO 568
C     -----REDUCE PBAR ERROR TEST-----
  567 ERROR=.9*ERROR
      NSAME=0
      WRITE (69,13) ERROR
      GO TO 300
  568 IF (INCOMP.EQ.1) GO TO 632
C     -----CALCULATE ENERGY-----
  600 DO 615 J=2,JMIN
      JPP=J+2
      JP=J+1
      JM=J-1
  601 DO 614 I=2,IMIN
      IP=I+1
      IM=I-1
  602 R1=RAT1(I)
      R2=RAT2(I)
C 603 U=UR(I,J)
C     UIPP=UR(I+2,J)
C     UIP=UR(IP,J)
C     UIM=UR(IM,J)
C     UJP=UR(I,JP)
C     UJM=UR(I,JM)
C     UIPJP=UR(IP,JP)
C     UIPJM=UR(IP,JM)
C 604 V=VZ(I,J)
C     VIP=VZ(IP,J)
C     VIM=VZ(IM,J)
C     VJPP=VZ(I,JPP)
C     VJP=VZ(I,JP)
C     VJM=VZ(I,JM)
C     VIPJP=VZ(IP,JP)
C     VIMJP=VZ(IM,JP)
C 605 UU=UIP*U
C     VV=VJP*V
C 606 S=SIG(I,J)
C     E=ENG(I,J)
C     QIJ=Q(I,J)
```

```
      PBAR=PB(I,J)+2.
C  607 Q1=QIJ+Q(IP,J)
C     Q2=QIJ+Q(IM,J)
C     Q3=QIJ+Q(I,JP)
C     Q4=QIJ+Q(I,JM)
  608 FIRST=SOLD(I,J)*ECLD(I,J)+CC21*(R2*(S+SIG(IM,J))*(E+ENG(IM,J))*U-R1-
     Z    *(S+SIG(IP,J))*(E+ENG(IP,J))*UIP+CC2*((S+SIG(I,JM))*(E+ENG(I,JM))-
     Z    *V-(S+SIG(I,JP))*(E+ENG(I,JP))*VJP))
  609 SECOND=CC22*(R2*(PBAR+PB(IM,J))*U-R1*(PBAR+PB(IP,J))*UIP+CC2-
     Z    *((PBAR+PB(I,JM))*V-(PBAR+PB(I,JP))*VJP))
  610 THIRD=CC23*(R1*(-Q1*UIP+4.*UIP*(UR(I+2,J)-U)+2.*(VIPJP*VIP-VV)-
     Z    +CC2*(VIPJP+VJP+VIP+V)*(UIPJP-UIPJM))+R2*(Q2*U-4.*U*(UIP-UIM)-2.-
     Z    *(VV-VIMJP*VIM)-CC2*(VJP+VIMJP+V+VIM)*(UJP-UJM))+CC25*(GAMMY(I,J)-
     Z    +1.)/NPR(I,J)*(R1*(EYE(IP,J)-EYE(I,J))-R2*(EYE(I,J)-EYE(IM,J))))-
     Z    /NRE(I,J)
  611 FOURTH=CC24*(-Q3*VJP+CC5*((UIPJP*UJP-UU)+2.*VJP*(VZ(I,JPP)-V))-
     Z    +.5*(UIPJP+UJP+UIP+U)*(VIPJP-VIMJP)+Q4*V-CC5*((UU-UIPJM*UJM)-
     Z    +2.*V*(VJP-VJM))-.5*(UIP+U+UIPJM+UJM)*(VIP-VIM)+CC25*(GAMMY(I,J)-
     Z    +1.)/NPR(I,J)*(EYE(I,JP)+EYE(I,JM)-2.*EYE(I,J)))/NRE(I,J)
  612 ENEW(I,J)=(FIRST+SECOND+THIRD+FOURTH)/S
  613 EYE(I,J)=ENEW(I,J)-.125*((UIP+U)**2+(VJP+V)**2)
  614 CONTINUE
  615 CONTINUE
C     -----RESTORE CURRENT ENERGY ARRAY-----
  620 DO 623 J=2,JMIN
  621 DO 622 I=2,IMIN
      ENG(I,J)=ENEW(I,J)
  622 CONTINUE
  623 CONTINUE
C----------ENERGY BCS & INCREASE ERROR TEST-----
  630 CALL BOUND
  631 NOUTER=NOUTER+1
      IF (NOUTER.GE.NOUT) GO TO 632
      GO TO 270
  632 IF (NCONT.EQ.0) GO TO 633
      NSAME=0
      GO TO 635
  633 NSAME=NSAME+1
      IF (NSAME.LT.5) GO TO 635
  634 ERROR=ERROR/.9
      NSAME=0
      WRITE (69,13) ERROR
  635 IF (NFIRST.EQ.0) GO TO 700
  636 WRITE (14) PB
      REWIND 14
      NFIRST=0
C     -----UPDATE END-OF-CYCLE VALUES-----
  700 DO 706 J=1,JMAX
      VZOLD(1,J)=VZ(1,J)
      UROLD(IMAX,J)=UR(IMAX,J)
  701 DO 705 I=2,ILES
  702 VZOLD(I,J)=VZ(I,J)
      IF (J.EQ.JMAX) GO TO 705
      UROLD(I,J)=UR(I,J)
```

```
  703 ECLD(I,J)=ENG(I,J)
      EYCLD(I,J)=EYE(I,J)
  704 SCLD(I,J)=SIG(I,J)
  705 CONTINUE
  706 CONTINUE
C-----------CALC COMPONENT FLUX AND REACTION-----
  709 IF (NSPECY.GT.1) CALL MASS
      CALL PROPC
      CALL BCUNF
C     -----CALCULATE NEW PRESSURE-----
  710 IF (INCOMP.EQ.0) GO TO 715
  711 DO 714 J=2,JMIN
  712 DO 713 I=2,IMIN
      PR(I,J)=PR(I,J)
  713 CONTINUE
  714 CONTINUE
      GO TO 719
  715 DO 718 J=2,JMIN
  716 DO 717 I=2,IMIN
      PR(I,J)=GAMMY(I,J)*SIG(I,J)*EYE(I,J)/EULNO-1.
  717 CONTINUE
  718 CONTINUE
C     -----PR BCS-----
  719 CALL BCUNE
C     -----WRITE OUTPLT & STORE FOR PLOTS-----
  750 IF (MOD(NCOUNT,NTRITE).NE.0) GO TO 800
      IF(NRITE.EQ.0.AND.NTAPE.EQ.0) GO TO 800
      NT=NT+1
  760 CALL RITER
      NSAVE=1
      NCOUNT=0
C     -----ADJUSTMENT OF TIME STEP FROM TERMINAL-----
  800 CALL RESET
      IF (NRESET.EQ.0) GO TO 801
 8000 IF (NTLOOP.LT.3) GO TO 801
      NRESET=0
 8001 CC6=THETA*CC6
      CC7=EULNO*(1.-PHI)
      CC12=EULNO*PHI
      CC13=RDL*CC12
      CC14=RDL*CC7
      CC19=THETA/DSQIG
      CC20=(1.-THETA)/DSQIG
 8002 IF (NDTAU.EQ.0) GO TO 804
  801 IF (NDTAU.EQ.0) GO TO 900
      NDTAU=0
  802 RATDT=DTAU/DTOLD
  803 CC6=CC6*RATDT*RATDT
      CC9=CC9*RATDT
      CC10=CC10*RATDT
      CC11=CC11*RATDT
      CC18=CC18/RATDT
      CC21=CC21*RATDT
      CC22=CC22*RATDT
```

```
      CC23=CC23*RATDT
      CC24=CC24*RATDT
      CC26=CC26*RATDT*NIN/NINOLD
      CC27=CC26*RDLSQ
      CC28=CC28*RATDT
      CC29=CC29*RATDT
      CC30=CC30*RATDT
      CC31=CC31*RATDT
      CC32=CC32/RATDT*NIN/NINOLD
      CC33=CC32/CC2
      CC34=CC33+CC33
      CC38=CC28*RATDT*NINOLD/NIN*NYOLD/NY
  804 CC40=CC6*CC12
      CC41=2./DCHI
      CC42=.5*CC40*RDLSQ
  805 BI1=-CC40-CC41
      BI2=BI1-4.*CC41
      AJ=CC42
      BJ1=-2.*CC42-CC41
      BJ2=BJ1-4.*CC41
      CJ=CC42
  806 DO 808 I=2,ILES
  807 AI(I)=CC40*RAT2(I)
      CI(I)=CC40*RAT1(I)
  808 CONTINUE
C     -----WRITE RESTART DATA AND EXIT WITH FINAL WRITEOUT-----
  900 IF (MOD(NTLOOP,NSAVE).NE.0) GO TO 902
  901 WRITE (10) PB,PR,VZ,UP,SOUN,ENG,EYE,SIG,DTAU,TCYC,ROUT,RDL,PHI,-
 Z THETA,PSI,BETAM,BETAV,BETAI,BETAK,T,DCHI,ERRCP,TAU,TAUEND,VUP,VREF,-
 Z PUP,PUP2,SIGUP,EY,VIN,VOUT,CELLIN,RIN,FRACP,JSTART,ITMAX,NTRITE,-
 Z NPRITE,NR,NL,NPROB,NFLU,IACCMP,NCLT,NRUN,NT,NNN,NTUBE,NANN,NSAVE,-
 Z NRITE,NTAPE,NTLOOP,NCOUNT,NREACT,NY,NIN,NSPECY,(ITERS(I),I=1,NNN)
      IF (NSPECY.EQ.1) GO TO 9010
      WRITE (10) NRE,NPR,GAMMY,MOLMIX,CONC,SIGK,MOLFR,MLES,GAMIN,-
 Z    GAMOW,TEMPOW,SIZEX,MMIXIN,MMIXOW,MLESIN,MLESOW,MOLFUP,MMIXUP,-
 Z    MLESUP,SMITUP,GAMUP,SMITOW,REYNIN,REYNOW,REYUP,SIGKUP,SKUP,-
 Z    NSC,EYEOW,ENGOW,SIGOW,TEOW
 9010 REWIND 10
      ENDFILE 6
      BACKSPACE 6
      IF (NTAPE.EQ.0) GO TO 902
      ENDFILE 12
      BACKSPACE 12
      NSAVE=2000
  902 TTEST=1.1*(TAUEND-TAU)
      IF (TTEST.GT.DTAU) GO TO 200
  903 WRITE (6,7)
      WRITE (6,8) (ITERS(K),K=1,NNN)
  904 CALL CPUTIM(MLSEC)
      SECTWO=MLSEC
      SEC=(SECTWO-SECONE)/1000.
      WRITE (69,1) SEC
C     STOP
C     END
```

```
C NIT12 - WIEBER SUBROUTINE INIT FOR FMIX5 - PRESUMES A TUBE WITH
C         COAXIAL ENTRY [illegible] A COMPRESIBLE FLUID. THE CENTER TUBE
C         COMPOSITION IS [illegible] THE ANNULAR COMPOSITION OF SPECIES CO,
C         C2, CC2, H2C, AND N2 IS READ IN, AS IS THE TEMPERATURE OF THE
C         ANNULAR STREAM. INPUT VELOCITES, COMPOSITIONS, AND
C         TEMPERATURES ARE HELD CONSTANT. OUTPUT PRESSURE IS HELD
C         CONSTANT, HENCE PROBLEM IS VALID ONLY FOR SUBSONIC FLOWS.
C         FLUID IS CONFINED BY IMPERMEABLE NO-SLIP WALLS.
C
C
      SUBROUTINE INIT
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION PR(23,43),RHO(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
     Z EYE(23,43),ENG(23,43)
      DIMENSION GOLD(23,43),G(23,43)
      DIMENSION B(23,43),D(23,43),SOLD(23,43),SKUP(23,5)
      DIMENSION SCUN(23,43),CONT(23,43)
      DIMENSION C(23,43),ENEW(23,43),SMITIN(4),REYUP(23)
      DIMENSION GOLD(23,43),EYOLD(23,43),VZOLD(23,43),UROLD(23,43),EOLD(23,43)
      DIMENSION BETAMT(23,43),BETAVR(23,43),BETAVZ(23,43)
      DIMENSION SIGK(23,43,5),T(23,43)
      DIMENSION CONC1(23,43),CONC2(23,43),CONC3(23,43),CONC4(23,43),-
     Z CONC5(23,43)
      DIMENSION SIGKUP(23,5),SMITUP(23,4),GAMUP(23),SMITOW(4),SKINIT(5)
C
      REAL*8 MLSTAR,MSTAR,LT,MOLWT(5),MOLMIX(23,43),SIZEV(3),SIZEX(6)
      REAL*8 MSC(23,43,4),MFR(23,43,5),MOLFUP(23,5),MLES(23,43,4)
      REAL*8 MRE(23,43),MPE(23,43),GAMMY(23,43),CONC(23,43)
      REAL*8 MMIXIN,MMIXOW,MLESIN(4),MLESOW(4),MFINIT(5)
      REAL*8 MMIXUP(23),MLESUP(23,4)
C
      INTEGER SPECIE(6)
C
      NAMELIST/SIZES/SIZEV
      NAMELIST/SIZEC/SIZEX
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/BB/EYOLD,VZOLD,UROLD,EOLD
      COMMON/CC/PR,SIG,UR,VZ,EYE,ENG
      COMMON/DD/BC4,BC6,ERRS,ITEST,JTEST,NSTOP,NCOUNT,ERROR1
      COMMON/EE/PB,PTIME,TAU
      COMMON/FF/GOLD,G
      COMMON/GG/PUP,PUPMAX,SIGUP,SUPMAX,DSCIG,VELUP,VUPMAX,RDL,FREQ
      COMMON/HH/B,D,SOLD,SUP2,SUP2
      COMMON/KK/SCUN,CONT
      COMMON/LL/EYSTAR,RGUT,LT,EULNO,REYNO,STRONO,SMITNO,PRANNO,VREF
      COMMON/MM/C,ENEW
      COMMON/NN/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/QQ/RKCOEF,CC15,CC17,CC32,CC33,CC34,CC35,CC36,CC37
      COMMON/TT/GOLD,BETANT,BETAVR,BETAVZ,JSTART
      COMMON/XX/RIN,VIN,VOLT,NTUBE,NANN
      COMMON/AB/MSTAR,TSTAR,CSTAR,DVSTAR,CC38,SPECIE
```

```
      COMMON/AC/ARE,AFR,GAMMY,MOLMIX,CONC
      COMMON/AD/RSC
      COMMON/AF/SIGK,MOLER
      COMMON/AG/CONC1,CONC2,CONC3,CONC4,CONC5
      COMMON/AH/SIGKUP,SKUP
      COMMON/AL/MLES,T,GAMIN,GAMOW,TEMPOW,SIZEX
      COMMON/AM/MOLFUP,MMIXUP,MLESUP,SMITUP,GAMUP,SMITIN,SMITOW,-
     Z    REYNIN,REYNOW,REYUP
      COMMON/AN/MMIXIN,MMIXOW,MLESIN,MLESOW
      COMMON/AO/EYEOW,ENGOW,SIGOW,TPOW
C
    1 FORMAT(T2,'READ IN DIMENSIONAL INPUT VELOCITIES AND LAST CELL OF-
     Z INSIDE TUBE AS SIZES: SIZEV=VIN,VOUT,CELLIN')
    2 FORMAT(T2,'VIN='F 8.4,'  VOUT='F8.4,'  CELLIN='F3.0)
    3 FORMAT (T2,'READ ANNULAR MOLE FRACTIONS AS SIZES: SIZEX=XCO, XO2,-
     Z XCO2, XH2O, XN2, TEMPERATURE')
    4 FORMAT (T2,'CO='F5.3,' O2='F5.3,' CO2='F5.3,' H2O='F5.3,' N2='F5.3,-
     Z    ' TEMP='F7.2)
    5 FORMAT(T3,'SOCKETUUM')
C
C     -----READ IN INPUT VELOCITES AND LAST CENTER TUBE CELL-----
  100 WRITE (69,1)
      READ (5,SIZES)
      VIN=SIZEV(1)
      VOUT=SIZEV(2)
      CELLIN=SIZEV(3)
  101 WRITE (69,2) VIN,VOUT,CELLIN
      NTUBE=CELLIN
  102 RIN=ROUT*NTUBE/IMAX
      VREF=VOUT+(VIN-VOUT)*RIN*RIN/(ROUT*ROUT)
      VIN=VIN/VREF
      VOUT=VOUT/VREF
      VUP=VIN/VREF
      VUPMAX=VOUT/VREF
      EY=EYSTAR/(VREF*VREF)
      NTUBE=NTUBE+1
      NANN=NTUBE+1
      EULNO=PSTAR/(SGSTAR*VREF*VREF)
      REYNO=SGSTAR*VREF*ROUT/MUSTAR
      PARTRE=REYNO*MUSTAR
  103 PUP=0.
      PUPMAX=0.
      SIGUP=1.
      SUPMAX=1.
      INCOMP=0
C----------SET CENTER TUBE MOLE FRACTIONS-----
 1030 DO 1031 I=2,NTUBE
      MOLFUP(I,1)=0.
      MOLFUP(I,2)=0.42
      MOLFUP(I,3)=0.
      MOLFUP(I,4)=0.
      MOLFUP(I,5)=1.58
 1031 CONTINUE
C----------READ IN ANNULAR CONCENTRATIONS AND TEMPERATURE-----
```

```
  104 WRITE (69,3)
      READ (5,SIZEC)
      WRITE (69,4) SIZEX
C----------SET ANNULAR UPSTREAM MOLE FRACTIONS AND TEMPERATURE-----
  105 DO 106 K=1,5
      DO 106 I=NANN,IMIN
      MOLEUP(I,K)=SIZEX(K)+SIZEX(K)
  106 CONTINUE
      TEMPOW=2.*(SIZEX(6)/TSTAR-1.)
      TPOW=TEMPOW/2.
C----------SET UPSTREAM MIXTURE MOLECULAR WEIGHTS-----
  107 MMIXIN=.42*MOLWT(2)+1.58*MOLWT(5)
      MMIXOW=2.*(SIZEX(1)*MOLWT(1)+SIZEX(2)*MOLWT(2)+SIZEX(3)*-
Z     MOLWT(3)+SIZEX(4)*MOLWT(4)+SIZEX(5)*MOLWT(5))
 1070 DO 1071 I=2,NTUBE
      MMIXUP(I)=MMIXIN
 1071 CONTINUE
 1072 DO 1073 I=NANN,IMIN
      MMIXUP(I)=MMIXOW
 1073 CONTINUE
  108 MLESIN(1)=MMIXIN
      MLESIN(2)=2.*MOLWT(5)
      MLESIN(3)=MMIXIN
      MLESIN(4)=MMIXIN
 1080 DO 1081 K=1,4
      MLESOW(K)=(MMIXOW-2.*SIZEX(K)*MOLWT(K))/(1.-SIZEX(K))
 1081 CONTINUE
 1082 DO 1087 K=1,4
 1083 DO 1084 I=2,NTUBE
      MLESUP(I,K)=MLESIN(K)
 1084 CONTINUE
 1085 DO 1086 I=NANN,IMIN
      MLESUP(I,K)=MLESOW(K)
 1086 CONTINUE
 1087 CONTINUE
C----------CALL IN MORE GAS PROPERTIES-----
  109 CALL PROPA
      EYEOW=2.*8.31434D+7*SIZEX(6)/(GAMOW*MMIXOW*MSTAR*VREF*VREF)
      ENGOW=EYEOW+.5*VOUT*VOUT
      SIGOW=EULAC/(GAMOW*EYEOW)
C--------SET UPSTREAM SCHMIDT NUMBERS-----
 1090 DO 1095 K=1,4
 1091 DO 1092 I=2,NTUBE
      SMITUP(I,K)=2.*SMITIN(K)
 1092 CONTINUE
 1093 DO 1094 I=NANN,IMIN
      SMITUP(I,K)=2.*SMITOW(K)
 1094 CONTINUE
 1095 CONTINUE
C----------SET UPSTREAM REYNOLDS NUMBERS-----
 1096 DO 1097 I=2,NTUBE
      REYUP(I)=REYNIN+REYNIN
 1097 CONTINUE
 1098 DO 1099 I=NANN,IMIN
```

```
      REYLP(I)=REYNCW*REYNCW
 1099 CONTINUE
C----------SET ZERO VARIABLES IN ENTIRE GRID-----
  110 DO 118 J=1,JMAX
  111 DO 117 I=1,IMAX
  112 PR(I,J)=0.
      PB(I,J)=0.
      UR(I,J)=0.
      VZ(I,J)=0.
      URCLD(I,J)=0.
      VZOLD(I,J)=0.
  113 CONT(I,J)=0.
      BETAMT(I,J)=0.
      BETAVR(I,J)=0.
      BETAVZ(I,J)=0.
  114 C(I,J)=0.
      B(I,J)=0.
      D(I,J)=0.
      G(I,J)=0.
  115 QOLD(I,J)=0.
      QCLD(I,J)=0.
      T(I,J)=0.
C----------SET NONZERO VARIABLES IN ENTIRE GRID-----
  116 ENG(I,J)=EY
      ENEW(I,J)=EY
      EYE(I,J)=EY
      ECLD(I,J)=EY
      EYCLD(I,J)=EY
      NRE(I,J)=REYNO
      NPR(I,J)=PRANBC
      GAMMY(I,J)=GAMIN
      MOLMIX(I,J)=1.
      SCON(I,J)=GAMMY(I,J)*EY/EULNO
      SIG(I,J)=1.
      SOLD(I,J)=1.
      CONC(I,J)=1.
  117 CONTINUE
  118 CONTINUE
C----------DEFINE SPECIE QUANTITIES FOR ENTIRE GRID-----
  120 MFINIT(1)=0.
      MFINIT(2)=0.21
      MFINIT(3)=0.
      MFINIT(4)=0.
      MFINIT(5)=0.79
  121 SKINIT(1)=0.
      SKINIT(2)=0.21*MOLWT(2)
      SKINIT(3)=0.
      SKINIT(4)=0.
      SKINIT(5)=0.79*MOLWT(5)
      WRITE (69,5)
C----------STORE IN GRID-----
  130 DO 132 K=1,5
      XX=MFINIT(K)
  131 DO 132 J=1,JLES
```

```
      DO 132 I=1,ILES
      MOLFR(I,J,K)=XX
  132 CONTINUE
  133 DO 135 K=1,5
      XX=SKINIT(K)
  134 DO 135 J=1,JLES
      DO 135 I=1,ILES
      SIGK(I,J,K)=XX
  135 CONTINUE
  136 DO 137 J=1,JLES
      DO 137 I=1,ILES
      CONC1(I,J)=0.
      CONC2(I,J)=0.21
      CONC3(I,J)=0.
      CONC4(I,J)=0.
      CONC5(I,J)=0.79
  137 CONTINUE
  138 DO 139 K=1,4
      XX=.5*MLESIN(K)
      DO 139 J=1,JLES
      DO 139 I=1,ILES
      MLES(I,J,K)=XX
  139 CONTINUE
C----------LOAD INPUT VELOCITIES-----
  140 DO 141 I=2,NTUBE
      VZ(I,2)=VIN
  141 CONTINUE
  142 DO 143 I=NANN,IMIN
      VZ(I,2)=VOUT
  143 CONTINUE
C----------SPECIFY INPUT MASS CONCENTRATIONS-----
  150 DO 155 K=1,5
      XX=MOLWT(K)/MWIXIN
  151 DO 152 I=2,NTUBE
      SIGKUP(I,K)=MOLFUP(I,K)*XX
  152 CONTINUE
      XX=MOLWT(K)*SIGDW/MWIXDW
  153 DO 154 I=NANN,IMIN
      SIGKUP(I,K)=MOLFUP(I,K)*XX
  154 CONTINUE
  155 CONTINUE
C----------SET SOME UPSTREAM CONSTANTS-----
  160 DO 161 I=2,NTUBE
      GAMUP(I)=2.*GAMIN
  161 CONTINUE
  162 DO 163 I=NANN,IMIN
      GAMUP(I)=2.*GAMDW
  163 CONTINUE
C
  200 JSTART=3
      INCOMP=0
      RETURN
      END
```

```
C  BOUNS - WIEBER SUBROUTINE BOUN FOR FLOMIX - PRESUMES A DUCT WITH
C         COAXIAL FLOW ENTRY OF A COMPRESSIBLE FLUID WITHIN IMPERMEABLE
C         NO-SLIP WALLS. THE UPSTREAM PRESSURE IS CALCULATED USING THE
C         MOMENTUM EQUATIONS AND DOWNSTREAM PRESSURE IS HELD CONSTANT.
C         SUITABLE ONLY FOR LOW MACH NUMBERS. CENTER JET FLUID IS AIR
C         AT 70 DEG F AND ANNULAR FLUID IS AN IDEAL MIXTURE OF CO, O2,
C         CO2, H2O, AND N2 AT A DIFFERENT TEMPERATURE. THESE CONCENTRATIONS
C         ARE HELD CONSTANT. USED IN NPROP=13
C
C
      SUBROUTINE BOUN
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION PB(23,43),PR(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
     Z  EYE(23,43),ENG(23,43)
      DIMENSION B(23,43),C(23,43),SOLD(23,43),SKUP(23,5),SIZEX(6)
      DIMENSION RAT1(23),RAT2(23)
      DIMENSION SIGK(23,43,5),SCUN(23,43),CONT(23,43)
      DIMENSION CONCUP(23),CONUP1(23),CONUP2(23),CONUP3(23),CONUP4(23),-
     Z  CONUP5(23)
      DIMENSION SIGKUP(23,5),SMITUP(23,4),GAMUP(23),T(23,43),REYUP(23)
      DIMENSION SMITOW(4),SMITIN(4)
      DIMENSION CONC1(23,43),CONC2(23,43),CONC3(23,43),CONC4(23,43),-
     Z  CONC5(23,43)
C
      REAL*8 LT,MOLMIX(23,43),MOLFR(23,43,5),MOLFUP(23,5)
      REAL*8 NRE(23,43),NPR(23,43),GAMMY(23,43),CONC(23,43)
      REAL*8 MUSTAR,MOLWT(5),MLESUP(23,4),MMIXUP(23)
      REAL*8 MLES(23,43,4),NSC(23,43,4)
      REAL*8 KVIN,KVOUT
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/CC/PR,SIG,UR,VZ,EYE,ENG
      COMMON/DD/BC4,BC6,ERRS,ITEST,JTEST,NSTOP,NCOUNT,ERROR1
      COMMON/EE/PB,PTIME,TAU
      COMMON/GG/PUP,PUPMAX,SIGUP,SUPMAX,DSGIG,VELUP,VUPMAX,RDL,FREQ
      COMMON/HH/B,O,SOLD,PUP2,SUP2
      COMMON/KK/SCUN,CONT
      COMMON/LL/EYSTAR,ROUT,LT,EULNO,REYNO,STRONO,SMITNO,PRANNO,VREF
      COMMON/NN/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/OO/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAI,BETAK,DCHI
      COMMON/WW/RAT1,RAT2
      COMMON/XX/RIN,VIN,VOUT,NTUBE,NANN
      COMMON/AC/NRE,NPR,GAMMY,MOLMIX,CONC
      COMMON/AD/NSC
      COMMON/AE/CC2,CC8,CC9,CC16,NPROP,NSPECY,NSPEC,MULTID,NY,NIN,NREACT
      COMMON/AF/SIGK,MOLFR
      COMMON/AG/CONC1,CONC2,CONC3,CONC4,CONC5
      COMMON/AH/SIGKUP,SKUP
      COMMON/AL/MLES,T,GAMIN,GAMOW,TEMPOW,SIZEX
      COMMON/AM/MOLFUP,MMIXUP,MLESUP,SMITUP,GAMUP,SMITIN,SMITOW,-
     Z  REYMIN,REYMOW,REYUP
      COMMON/AO/EYEOW,ENGOW,SIGOW,TPOW
```

```
C
C----------CALLED AT FMIX6 #130-----
C      -----CALCULATE BC# COEFFICIENTS-----
   90 XI=ILES
      CCC=(XI+XI-3.)/(XI-1.)
   91 XI=IMIN
   92 BC1=CCC*RDL
      BC2=CCC*(XI-2.)/(XI+XI-3.)
      BC3=4./RDL
      BC4=RDL*EULNO
      BC5=2./RDL
      BC6=1./EULNO
      BC7=4.*EULNO
      BC8=8.*EULNC
   93 KVIN=2.*VIN*1.
      KVOUT=2.*VOUT*SIGOW
      EY=EYSTAR/(VREF*VREF)
      NFIRST=1
   94 DO 95 I=2,IMIN
      GAMMY(I,1)=GAMUP(I)-GAMMY(I,2)
   95 CONTINUE
C
      ENTRY BCUNA
C
C----------CALLED NEAR FMIX6 #232-----
C      -----CALCULATE PB, PR, UPSTREAM AND DOWNSTREAM-----
  100 DO 101 I=2,IMIN
      PR(I,1)=PR(I,2)-D(I,2)/BC4
      PB(I,JLES)=-PB(I,JMIN)
C  101 CONTINUE
C      -----WALL-----
C  102 DO 103 J=2,JLES
      PB(ILES,J)=PB(IMIN,J)+P(ILES,J)/EULNC
C  103 CONTINUE
      IF (NFIRST.EQ.1) GO TO 200
      IF (PHI.EQ.1.) RETURN
  104 DO 105 I=2,IMIN
      PR(I,1)=PR(I,2)-D(I,2)/BC4
  105 CONTINUE
  106 DO 107 J=2,JMIN
      PR(ILES,J)=PR(IMIN,J)+P(IMIN,J)/EULNC
  107 CONTINUE
      IF (NFIRST.EQ.1) GO TO 200
      RETURN
CC
C      ENTRY BCUNB
CC
C----------CALLED AT FMIX6 #410-----
C      -----CALCULATE SIG & SOLD, UPSTREAM AND DOWNSTREAM-----
  200 DO 201 I=2,IMIN
      SIG(I,1)=-SIG(I,2)+SOLD(I,1)+SOLD(I,2)+2.*(PB(I,1)+PB(I,2)-
     Z-PR(I,1)-PR(I,2))/(SOUN(I,1)+SOUN(I,2))
      SIG(I,JLES)=2.*SIG(I,JMIN)-SIG(I,JMN1)
C  201 CONTINUE
```

```
C----------WALL AND CENTERLINE-----
  202 DO 203 J=2,JMIN
      SIG(1,J)=SIG(2,J)
      SIG(ILES,J)=SIG(IMIN,J)
  203 CONTINUE
      IF (NFIRST.EQ.0) RETURN
  204 DO 205 I=NAAA,IMIN
      SIG(I,1)=2.*SIGOW-SIG(I,2)
  205 CONTINUE
C
      ENTRY BCUNC
C
C----------CALLED AT FMIX6 #460-----
C     -----CALCULATE VELOCITIES, UPSTREAM AND DOWNSTREAM-----
  300 DO 301 I=2,NTUBE
      VZ(I,2)=KVIN/(SIG(I,1)+SIG(I,2))
  301 CONTINUE
  302 DO 303 I=NAAA,IMIN
      VZ(I,2)=KVOUT/(SIG(I,1)+SIG(I,2))
  303 CONTINUE
  304 DO 307 I=2,IMIN
  305 VZ(I,1)=2.*VZ(I,2)-VZ(I,3)-BC3*(RAT1(I)*UR(I+1,2)-RAT2(I)*UR(I,2))
      UR(I,1)=-UR(I,2)
  306 VZ(I,JMAX)=2.*VZ(I,JLES)-VZ(I,JMIN)
      UR(I,JLES)=UR(I,JMIN)
  307 CONTINUE
C     -----WALL AND CENTERLINE-----
  308 DO 310 J=2,JLES
      VZ(ILES,J)=-VZ(IMIN,J)
      UR(IMAX,J)=-UR(IMIN,J)
  309 VZ(1,J)=VZ(2,J)
  310 CONTINUE
      IF (NFIRST.EQ.1) GO TO 400
      RETURN
C
      ENTRY BCUAD
C
C----------CALLED AT FMIX6 #630-----
C----------CENTERLINE AND WALL-----
  400 DO 401 J=2,JMIN
      EYE(1,J)=EYE(2,J)
      EYE(ILES,J)=EYE(IMIN,J)
  401 CONTINUE
C----------DOWNSTREAM-----
  402 DO 403 I=2,IMIN
      EYE(I,JLES)=BCP/((GAMMY(I,JMIN)+GAMMY(I,JLES))*(SIG(I,JMIN)-
Z     +SIG(I,JLES)))-EYE(I,JMIN)
      ENG(I,JLES)=EYE(I,JMIN)+EYE(I,JLES)+VZ(I,JLES)**2+UR(I,JMIN)**2-
Z     -ENG(I,JMIN)
  403 CONTINUE
C----------UPSTREAM-----
  404 DO 405 I=2,NTUBE
      EYE(I,1)=2.*EY*(.5*(SIG(I,1)+SIG(I,2)))**GAMIN-EYE(I,2)
      EIN=EYE(I,1)+EYE(I,2)+VZ(I,2)**2
```

```
      ENG(I,1)=EIN-ENG(I,2)
  405 CONTINUE
  406 DO 407 I=NANN,IMIN
      EYE(I,1)=2.*EYECW*(.5*(SIG(I,1)+SIG(I,2))/SIGCW)**GAMOW-EYE(I,2)
      EIN=EYE(I,1)+EYE(I,2)+VZ(I,2)**2
      ENG(I,1)=EIN-ENG(I,2)
  407 CONTINUE
      IF (NFIRST.EQ.1) GO TO 600
      RETURN
C
      ENTRY BCUNF
C
C----------CALLED NEAR MASS2 #250-----
C----------CALC UPSTREAM AND DOWNSTREAM MASS CONCENTRATIONS-----
  500 DO 502 K=1,4
  501 DO 502 I=2,IMIN
      SIGK(I,1,K)=2.*SKUP(I,K)-SIGK(I,2,K)
      SIGK(I,JLES,K)=2.*SIGK(I,JMIN,K)-SIGK(I,JMNI,K)
  502 CONTINUE
  503 DO 505 K=1,4
  504 DO 505 J=2,JMIN
      SIGK(1,J,K)=SIGK(2,J,K)
      SIGK(ILES,J,K)=SIGK(IMIN,J,K)
  505 CONTINUE
      RETURN
C
      ENTRY BCUNG
C
C----------CALLED AT MASS2 #212 AND NEAR EMIXE #201-----
C----------CALC UPSTREAM AND DOWNSTREAM QUANTITIES-----
C----------TRIPLE SUBSCRIPTS-----
  600 IF (NPROP.EQ.1) GO TO 602
      DO 601 K=1,5
 6000 DO 6001 I=2,NTUBE
      SKUP(I,K)=.5*SIGKUP(I,K)*(SIG(I,1)+SIG(I,2))
 6001 CONTINUE
 6002 DO 6003 I=NANN,IMIN
      SKUP(I,K)=SIGKUP(I,K)*.5*(SIG(I,1)+SIG(I,2))/SIGOW
 6003 CONTINUE
  601 CONTINUE
  602 DO 603 K=1,5
      DO 603 I=2,IMIN
      SIGK(I,1,K)=2.*SKUP(I,K)-SIGK(I,2,K)
      SIGK(I,JLES,K)=2.*SIGK(I,JMIN,K)-SIGK(I,JMNI,K)
  603 CONTINUE
  604 DO 605 K=1,5
      DO 605 I=2,IMIN
      MOLFR(I,1,K)=MOLFUP(I,K)-MOLFR(I,2,K)
      MOLFR(I,JLES,K)=2.*MOLFR(I,JMIN,K)-MOLFR(I,JMNI,K)
  605 CONTINUE
  606 DO 607 K=1,4
      DO 607 I=2,IMIN
      MLES(I,1,K)=MLESUP(I,K)-MLES(I,2,K)
      MLES(I,JLES,K)=2.*MLES(I,JMIN,K)-MLES(I,JMNI,K)
```

```
  607 CONTINUE
  608 DO 609 K=1,4
      DO 609 I=2,IMIN
      NSC(I,1,K)=SMITUP(I,K)-NSC(I,2,K)
      NSC(I,JLES,K)=2.*NSC(I,JMIN,K)-NSC(I,JMNI,K)
  609 CONTINUE
C----------DOUBLE SUBSCRIPTS-----
  610 DO 611 I=2,IMIN
      CONCUP(I)=2.*(SIG(I,1)+SIG(I,2))/MMIXUP(I)
  611 CONTINUE
  612 DO 613 I=2,IMIN
      MOLMIX(I,1)=MMIXUP(I)-MOLMIX(I,2)
      MOLMIX(I,JLES)=2.*MOLMIX(I,JMIN)-MOLMIX(I,JMNI)
      CONC(I,1)=CONCUP(I)-CONC(I,2)
      CONC(I,JLES)=2.*CONC(I,JMIN)-CONC(I,JMNI)
  613 CONTINUE
  614 DO 615 I=2,IMIN
      CONUP1(I)=2.*SKUP(I,1)/MOLWT(1)
      CONUP2(I)=2.*SKUP(I,2)/MOLWT(2)
      CONUP3(I)=2.*SKUP(I,3)/MOLWT(3)
      CONUP4(I)=2.*SKUP(I,4)/MOLWT(4)
      CONUP5(I)=2.*SKUP(I,5)/MOLWT(5)
  615 CONTINUE
  616 DO 617 I=2,IMIN
      CONC1(I,1)=CONUP1(I)-CONC1(I,2)
      CONC1(I,JLES)=2.*CONC1(I,JMIN)-CONC1(I,JMNI)
      CONC2(I,1)=CONUP2(I)-CONC2(I,2)
      CONC2(I,JLES)=2.*CONC2(I,JMIN)-CONC2(I,JMNI)
      CONC3(I,1)=CONUP3(I)-CONC3(I,2)
      CONC3(I,JLES)=2.*CONC3(I,JMIN)-CONC3(I,JMNI)
      CONC4(I,1)=CONUP4(I)-CONC4(I,2)
      CONC4(I,JLES)=2.*CONC4(I,JMIN)-CONC4(I,JMNI)
      CONC5(I,1)=CONUP5(I)-CONC5(I,2)
      CONC5(I,JLES)=2.*CONC5(I,JMIN)-CONC5(I,JMNI)
  617 CONTINUE
C----------CENTERLINE AND WALL-----
C----------TRIPLE SUBSCRIPTS-----
  630 DO 631 K=1,5
      DO 631 J=2,JMIN
      SIGK(1,J,K)=SIGK(2,J,K)
      SIGK(ILES,J,K)=SIGK(IMIN,J,K)
  631 CONTINUE
  632 DO 633 K=1,5
      DO 633 J=2,JMIN
      MOLFR(ILES,J,K)=MOLFR(IMIN,J,K)
  633 CONTINUE
  634 DO 635 K=1,4
      DO 635 J=2,JMIN
      MLES(1,J,K)=MLES(2,J,K)
      MLES(ILES,J,K)=MLES(IMIN,J,K)
  635 CONTINUE
  636 DO 637 K=1,4
      DO 637 J=2,JMIN
      NSC(ILES,J,K)=NSC(IMIN,J,K)
```

```
  637 CONTINUE
C---------DOUBLE SUBSCRIPTS-----
  640 DO 641 J=2,JMIN
      MOLMIX(1,J)=MOLMIX(2,J)
      MOLMIX(ILES,J)=MOLMIX(IMIN,J)
      CONC(1,J)=CONC(2,J)
      CONC(ILES,J)=CONC(IMIN,J)
  641 CONTINUE
  642 DO 643 J=2,JMIN
      CONC1(1,J)=CONC1(2,J)
      CONC1(ILES,J)=CONC1(IMIN,J)
      CONC2(1,J)=CONC2(2,J)
      CONC2(ILES,J)=CONC2(IMIN,J)
      CONC3(1,J)=CONC3(2,J)
      CONC3(ILES,J)=CONC3(IMIN,J)
      CONC4(1,J)=CONC4(2,J)
      CONC4(ILES,J)=CONC4(IMIN,J)
      CONC5(1,J)=CONC5(2,J)
      CONC5(ILES,J)=CONC5(IMIN,J)
  643 CONTINUE
      IF (NFIRST.EQ.1) GO TO 700
      RETURN

      ENTRY BOUNH

C---------CALLED NEAR FMIX6 #707 AND NEAR #201-----
C---------CALC GAMMY AND TEMPERATURE UPSTREAM AND DOWNSTREAM-----
  700 DO 701 I=2,IMIN
      GAMMY(I,1)=GAMUP(I)-GAMMY(I,2)
      GAMMY(I,JLES)=2.*GAMMY(I,JMIN)-GAMMY(I,JMNI)
      T(I,JLES)=(GAMMY(I,JMIN)+GAMMY(I,JLES))*(MOLMIX(I,JMIN)+MOLMIX-
     (I,JLES))*(EYE(I,JMIN)+EYE(I,JLES))/BC7-2.
  701 CONTINUE
  702 DO 703 I=2,NTUBE
      T(I,1)=2.*(.5*(SIG(I,1)+SIG(I,2)))**GAMIN-T(I,2)-2.
  703 CONTINUE
  704 DO 705 I=NAAA,IMIN
      T(I,1)=(TEMPOW+2.)*(.5*(SIG(I,1)+SIG(I,2))/SIGOW)**GAMOW-T(I,2)-2.
  705 CONTINUE
C---------CENTERLINE AND WALL-----
  714 DO 716 J=2,JMIN
  715 GAMMY(1,J)=GAMMY(2,J)
      GAMMY(ILES,J)=GAMMY(IMIN,J)
      T(1,J)=T(2,J)
      T(ILES,J)=T(IMIN,J)
  716 CONTINUE
C---------CALC REQUIRED BOUNDARY VALUES OF NRE-----
  717 DO 718 I=2,IMIN
      NRE(I,1)=REYUP(I)-NRE(I,2)
      NRE(I,JLES)=2.*NRE(I,JMIN)-NRE(I,JMNI)
  718 CONTINUE
  719 DO 720 J=2,JMIN
      NRE(ILES,J)=NRE(IMIN,J)
  720 CONTINUE
```

```
      IF (NFIRST.EQ.0) RETURN
C
      ENTRY BCUNE
C
C----------CALLED AT FMIX6 #719-----
C     -----CALCULATE PR UPSTREAM AND DOWNSTREAM-----
  800 DO 804 I=2,IMIN
  801 PR(I,1)=(GAMMY(I,1)+GAMMY(I,2))*(SIG(I,1)+SIG(I,2))*(EYE(I,1)+-
Z     EYE(I,2))/BC7-PR(I,2)-2.
      PR(I,JLES1)=-PR(I,JMIN)
  804 CONTINUE
C     -----WALL-----
  805 DO 806 J=2,JMIN
      PR(ILES,J)=(GAMMY(IMIN,J)+GAMMY(ILES,J))*(SIG(IMIN,J)+SIG(ILES,J))-
7     *(EYE(IMIN,J)+EYE(ILES,J))/BC7-PR(IMIN,J)-2.
  806 CONTINUE
      NFIRST=0
      RETURN
      END
```

```
C PROPS6 - WIEBER SUBROUTINE PROP FOR PROPERTIES OF FLUID #6, A MIXTURE
C          OF CARBON MONOXIDE, OXYGEN, CARBON DIOXIDE, WATER VAPOR, AND
C          OXYGEN WITH TEMPERATURES VARYING FROM 294 TO 2000 DEG K.
C          PRESSURE=ABOUT 1 ATM, MIXTURES AND GASES ASSUMED IDEAL.
C          COMPUTES SPECIFIC HEATS, GAMMA, VISCOSITIES, THERMAL
C          CONDUCTIVITIES, DIFFUSION COEFFICIENTS, AND
C          INTERNAL ENERGIES. COMPUTES REACTION RATE OF CO + O2 TO CO2
C          AS A FUNCTION OF TEMPERATURE.
C
C
      SUBROUTINE PROP
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION SIGK(23,43,5)
      DIMENSION CCTHER(6,5,2),COMU(3,5),COK(3,5),CODV(4,4)
      DIMENSION DV(4),SMITIN(4),SMITOW(4),XCP(5),REYUP(23),CP(23,43)
      DIMENSION XF(5),T(23,43),SIZEX(6),SMITUP(23,4),GAMUP(23)
      DIMENSION PR(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
Z     EYE(23,43),ENG(23,43)
C
      REAL*8 MUSTAR,MSTAR,LT,MOLWT(5),MOLMIX(23,43),MWT(5)
      REAL*8 NSC(23,43,4),MOLFR(23,43,5)
      REAL*8 NRE(23,43),NPR(23,43),GAMMY(23,43),CONC(23,43)
      REAL*8 MLWTOW,MMIXOW,MLWTIN,MMIXIN,MLESOW(4),MLESIN(4)
      REAL*8 NSC2(23,43),NSC3(23,43),NSC4(23,43),MLES(23,43,4)
      REAL*8 MOLFUP(23,5),MMIXUP(23),MLESUP(23,4)
C
      INTEGER SPECY1/4H  CO/,SPECY2/4H  O2/,SPECY3/4H CO2/,-
Z     SPECY4/4H H2O/,SPECY5/4H  N2/,SPECY6/4H AVG/
      INTEGER SPECIE(6)
C
      DATA CCTHER/3.7100928D+00,-1.6190964D-03,3.6923594D-06,-
Z     -2.0319674D-09,2.3953344D-13,-1.4356210D+04,3.6255985D+00,-
Z     -1.8782184D-03,7.0554544D-06,-6.7635137D-09,2.1555993D-12,-
Z     -1.0475226D+03,2.4007797D+00,8.7350957D-03,-6.6070878D-06,-
Z     2.2021861D-09,6.3274039D-16,-4.8377527D+04,4.0701275D+00,-
Z     -1.1084499D-03,4.1521160D-06,-2.9637404D-09,8.0702103D-13,-
Z     -3.0279722D+04,3.6748261D+00,-1.2081500D-03,2.3240102D-06,-
Z     -6.3217599D-10,-2.2577253D-13,-1.0611588D+03,2.9840696D+00,-
Z     1.4891390D-03,-5.7899684D-07,1.0364577D-10,-6.9353550D-15,-
Z     -1.4245228D+04,3.6219535D+00,7.3618264D-04,-1.9652228D-07,-
Z     3.6201558D-11,-2.8945627D-15,-1.2019825D+03,4.4608041D+00,-
Z     3.0981719D-03,-1.2392571D-06,2.2741325D-10,-1.5525954D-14,-
Z     -4.8961442D+04,2.7167633D+00,2.9451374D-03,-8.0224374D-07,-
Z     1.2266982D-10,-4.8472145D-15,-2.9905826D+04,2.8963194D+00,-
Z     1.5154866D-03,-5.7235277D-07,9.9807393D-11,-6.5223555D-15,-
Z     -9.0586184D+02/
C
      DATA COMU/7.0143939D-05,3.9449470D-07,-6.3708635D-11,-
Z     6.1640591D-05,5.2121698D-07,-1.0045311D-10,3.1294120D-05,-
Z     4.4C21098D-07,-8.0082151D-11,-4.1043305D-05,4.4367979D-07,-
Z     -2.8399462D-11,7.0065381D-05,3.9486838D-07,-6.3914576D-11/
C
```

```
      DATA CCK/1.4782635D-05,1.5677362D-07,-1.5274723D-11,-
     Z    5.1593532D-06,2.0251356D-07,-3.1030278D-11,-1.5141680D-05,-
     Z    1.9915205D-07,-2.8111257D-11,-3.6955975D-05,2.4435983D-07,-
     Z    6.5700666D-11,1.8609061D-05,1.5048485D-07,-1.1359894D-11/
C
      DATA CCOV/-7.4337571D-02,5.9744264D-04,1.0970252D-06,-
     Z    -6.1443812D-11,-9.5270522D-03,2.4471186D-04,1.6668767D-06,-
     Z    -2.6556088D-10,-7.0951581D-02,4.7988953D-04,1.1313812D-06,-
     Z    -1.1116494D-10,-1.7311585D-02,-1.3116238D-04,2.2191841D-06,-
     Z    -2.5229710D-10/
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/CC/PR,SIG,UR,VR,EYE,ENG
      COMMON/LL/EYSTAR,ROUT,LT,EULNO,REYNO,STRONO,SMITNO,PRANNO,VREF
      COMMON/NN/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/CC/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAI,BETAK,DCHI
      COMMON/AB/MSTAR,TSTAR,CSTAR,DVSTAR,CC38,SPECIE
      COMMON/AC/ARE,APR,GAMMY,MOLMIX,CONC
      COMMON/AD/ASC
      COMMON/AE/CC2,CC8,CC9,CC16,NPROF,NSPECY,NSPEC,MULTID,NY,NIN,NREACT
      COMMON/AF/SIGK,MOLFR
      COMMON/AL/MLES,T,GAMIN,GAMCH,TEMPCH,SIZEX
      COMMON/AM/MOLFUP,MMIXUP,MLESUP,SMITUP,GAMUP,SMITIN,SMITOH,-
     Z    REYNIN,REYNCH,REYUP
      COMMON/AN/MMIXIN,MMIXCH,MLESIN,MLESCH
C
C----------CALLED AT PMIX6 #130-----
C----------SET INITIAL VALUES-----
  100 MWT(1)=28.010
      MWT(2)=32.000
      MWT(3)=44.010
      MWT(4)=18.016
      MWT(5)=28.016
      SPECIE(1)=SPECY1
      SPECIE(2)=SPECY2
      SPECIE(3)=SPECY3
      SPECIE(4)=SPECY4
      SPECIE(5)=SPECY5
      SPECIE(6)=SPECY6
      NSPECY=5
      NSPEC=4
  101 MSTAR=.21*MWT(2)+.79*MWT(5)
      DO 102 K=1,5
      MOLWT(K)=MWT(K)/MSTAR
  102 CONTINUE
C----------CALCULATE REFERENCE CP AND GAMMA-1.-----
  103 TSTAR=5.*(70.-32.)/9.+273.1
  104 CPC2=1.9872*(COTHER(1,2,1)+TSTAR*(COTHER(2,2,1)+TSTAR*(COTHER(3,2,1)-
     Z    +TSTAR*(COTHER(4,2,1)+TSTAR*COTHER(5,2,1)))))
      CVC2=CPC2-1.9872
      CPN2=1.9872*(COTHER(1,5,1)+TSTAR*(COTHER(2,5,1)+TSTAR*(COTHER(3,5,1)-
     Z    +TSTAR*(COTHER(4,5,1)+TSTAR*COTHER(5,5,1)))))
      CVN2=CPN2-1.9872
      CPSTAR=(.21*CPC2+.79*CPN2)/MSTAR
```

```
      CVSTAR=(.21*CVO2+.79*CVN2)/MSTAR
      GAMIN=CPSTAR/CVSTAR-1.
C----------CALCULATE REFERENCE VISCOSITY AND THERMAL CONDUCTIVITY-----
  105 XMUO2=COMU(1,2)+TSTAR*(COMU(2,2)+TSTAR*COMU(3,2))
      XMUN2=COMU(1,5)+TSTAR*(COMU(2,5)+TSTAR*COMU(3,5))
      MUSTAR=.21*XMUO2+.79*XMUN2
      XKO2=COK(1,2)+TSTAR*(COK(2,2)+TSTAR*COK(3,2))
      XKN2=COK(1,5)+TSTAR*(COK(2,5)+TSTAR*COK(3,5))
      XKSTAR=.21*XKO2+.79*XKN2
  106 PSTAR=1.01325D+6
C----------R=8.31434D+7 (GM-SQ CM)/(SQ SEC-GM MOLE-DEG K)-----
      EYSTAR=8.31434D+7*TSTAR/(GAMIN*MSTAR)
      SGSTAR=PSTAR/(GAMIN*EYSTAR)
      CSTAR=SGSTAR/MSTAR
      CSMIT=MUSTAR/SGSTAR
C----------CALC DIFFUSIVITIES-----
  107 DO 108 K=1,4
      DV(K)=CODV(1,K)+TSTAR*(CODV(2,K)+TSTAR*(CODV(3,K)+TSTAR*-
     Z    CODV(4,K)))
  108 CONTINUE
      DVSTAR=DV(2)
      SMITNC=CSMIT/DVSTAR
      SMITIN(1)=CSMIT/DV(1)
      SMITIN(2)=CSMIT/DV(2)
      SMITIN(3)=CSMIT/DV(3)
      SMITIN(4)=CSMIT/DV(4)
  109 PRANNC=CPSTAR*MUSTAR/XKSTAR
      SMITNC=CSMIT/DVSTAR
      NPROP=0
      MULTID=1
  110 RATN=DSQRT(32.)/28.01
      RETURN
C
      ENTRY PROPA
C
C----------CALLED AT NIT12 #109-----
C----------CALC MU, XK, CP, CV, GAMMY, FOR ANNULUS AT START-----
  200 KT=1
      TP=SIZEX(6)
      CRE=ROUT*VREF*SGSTAR
      MLWTCW=MWIXOH/2.
      IF (TP.GT.1000.) KT=2
  201 SUMCP=0.
      SUMCV=0.
      SUMK=0.
      SUMM=0.
  202 DO 205 K=1,5
  203 XMU=COMU(1,K)+TP*(COMU(2,K)+TP*COMU(3,K))
      XXK=COK(1,K)+TP*(COK(2,K)+TP*COK(3,K))
      XCP(K)=1.9872*(CCTHER(1,K,KT)+TP*(CCTHER(2,K,KT)+TP*-
Z     (CCTHER(3,K,KT)+TP*(CCTHER(4,K,KT)+TP*CCTHER(5,K,KT)))))
      XCV=XCP(K)-1.9872
  204 SUMCP=SUMCP+XCP(K)*SIZEX(K)
      SUMCV=SUMCV+XCV*SIZEX(K)
```

```
      SUMK=SUMK+X*SIGFX(K)
      SUMM=SUMM+X*SIGFX(K)
  205 CONTINUE
C----------CALCULATE CONSTANT INPUT VALUES FOR ANNULUS-----
  206 GAMCW=SUMCP/SUMCV-1.
      CPCW=SUMCP/(MLWTOW*MSTAR)
  207 PRANCW=CPCW*SUMM/SUMK
      REYNOW=CRE/SUMM
      CSMIT=SUMM/SGSTAR
  210 DO 211 K=1,4
      DV(K)=CODV(1,K)+TP*(CODV(2,K)+TP*(CODV(3,K)+TP*CODV(4,K)))
      SMITCW(K)=CSMIT/DV(K)
  211 CONTINUE
C----------CALCULATE CONSTANT INPUT VALUES FOR CENTER JET-----
  220 TEMPIN=0.
      CPIN=CPSTAR
      PRANIN=PRANCW
      REYNIN=REYNC
      MLWTIN=1.
      RETURN
C
      ENTRY PROPR
C
C----------CALLED AT MASS2 #290-----
C----------CALC REACTION RATES-----
  300 DO 313 J=2,JMIN
  301 DO 312 I=2,IMIN
  302 T(I,J)=GAMMY(I,J)*MOLMIX(I,J)*EYE(I,J)/EULNC-1.
      TP=TEMP(I,J)*TSTAR
      TEMP1=SIGK(I,J,1)
      TEMP2=SIGK(I,J,2)
      TEMP3=SIGK(I,J,3)
      RATE=CC38*3.98E+9*DEXP(-1.15260+4/TP)
  303 DO 310 N=1,NY
      IF (TEMP1.LE.0..OR.TEMP2.LE.0.) GO TO 311
  304 DELK1=RATE*TEMP1*TEMP2
      DELK2=RATM*DELK1
      IF (DELK1.GT.TEMP1) GO TO 305
      IF (DELK2.GT.TEMP2) GO TO 306
      GO TO 308
  305 IF (DELK2.LE.TEMP2) GO TO 307
      TEST=TEMP2+TEMP2
      IF (TEMP1.LT.TEST) GO TO 307
  306 DELK2=TEMP2
      DELK1=TEMP2/RATM
      GO TO 308
  307 DELK1=TEMP1
      DELK2=RATM*DELK1
  308 DELK3=DELK1+DELK2
  309 TEMP1=TEMP1-DELK1
      TEMP2=TEMP2-DELK2
      TEMP3=TEMP3+DELK1+DELK2
  310 CONTINUE
  311 SIGK(I,J,1)=TEMP1
```

```
      SIGK(I,J,2)=TEMP2
      SIGK(I,J,3)=TEMP3
  312 CONTINUE
  313 CONTINUE
      RETURN
C
      ENTRY PROPC
C
C----------CALLED NEAR FMIX6 #201 AND FMIX6 #707-----
C----------CALC GRID PROPERTIES AT END OF TIME STEP BEFORE PRESSURE----------
C----------CALC NEW CP, CV, GAMMY, & TEMP-----
  400 DO 426 J=2,JMIN
  401 DO 425 I=2,IMIN
      KT=1
      TP=(T(I,J)+1.)*TSTAR
  402 IF (TP.GT.1000.) KT=2
  403 DO 404 K=1,5
      XF(K)=MOLFR(I,J,K)
  404 CONTINUE
C----------CALC MU, XK, CP, CV, GAMMY, & TEMP-----
  410 SUMCP=0.
      SUMCV=0.
      SUMK=0.
      SUMM=0.
  411 DO 414 K=1,5
      IF (XF(K).LT.1.D-7) GO TO 414
  412 XCP(K)=1.9872*(CCTHER(1,K,KT)+TP*(COTHER(2,K,KT)+TP*-
Z     (CCTHER(3,K,KT)+TP*(CCTHER(4,K,KT)+TP*CCTHER(5,K,KT)))))
      XCV=XCP(K)-1.9872
      XMU=COMU(1,K)+TP*(COMU(2,K)+TP*COMU(3,K))
      XXK=COK(1,K)+TP*(COK(2,K)+TP*COK(3,K))
  413 SUMK=SUMK+XXK*XF(K)
      SUMM=SUMM+XMU*XF(K)
      SUMCP=SUMCP+XCP(K)*XF(K)
      SUMCV=SUMCV+XCV*XF(K)
  414 CONTINUE
  415 GAMMY(I,J)=SUMCP/SUMCV-1.
      CP(I,J)=SUMCP/(MOLMIX(1,J)*MSTAR)
      T(I,J)=GAMMY(I,J)*MOLMIX(I,J)*EYE(I,J)/EULNC-1.
  416 CSMIT=SUMM/SQSTAR
      NRE(I,J)=CRE/SUMM
      NPR(I,J)=CP(I,J)*SUMM/SUMK
C----------CALC DV AND NSC-----
  420 DO 423 K=1,4
  421 DV(K)=CODV(1,K)+TP*(CODV(2,K)+TP*(CODV(3,K)+TP*CODV(4,K)))
  423 CONTINUE
  424 NSC(I,J,1)=CSMIT/DV(1)
      NSC2(I,J)=CSMIT/DV(2)
      NSC3(I,J)=CSMIT/DV(3)
      NSC4(I,J)=CSMIT/DV(4)
  425 CONTINUE
  426 CONTINUE
C----------STORE REST OF NSC NUMBERS-----
  430 DO 431 J=2,JMIN
```

```
      DO 431 I=2,IMIN
      NSC(I,J,2)=NSC2(I,J)
  431 CONTINUE
  432 DO 433 J=2,JMIN
      DO 433 I=2,IMIN
      NSC(I,J,3)=NSC3(I,J)
  433 CONTINUE
  434 DO 435 J=2,JMIN
      DO 435 I=2,IMIN
      NSC(I,J,4)=NSC4(I,J)
  435 CONTINUE
      NPROP=1
      RETURN
      END
```

```
C PADIE - NIEBER SUBROUTINE PADI FOR FLOMIX5 TO PERFORM THE ITERATIONS
C         OF THE PRESSURE FIELD - USES ALTERNATING DIRECTION IMPLICIT
C         (ADI) SCHEME OF BRIAN - INCLUDES TESTING AND INTERMEDIATE
C         WRITEOUT OF THE PRESSURE FIELD - PRESSURE UPSTREAM IS CALCULATED
C         USING MOMENTUM AND PRESSURE DOWNSTREAM IS HELD CONSTANT -
C         USED IN NPROB=13.
C
      SUBROUTINE PADI
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION GOLD(23,43),G(23,43)
      DIMENSION B(23,43),D(23,43),SOLD(23,43)
      DIMENSION PP(23,43),PP(23,43)
      DIMENSION AI(23),CI(23),DI(23),EI(23),FI(23),DJ(43),EJ(43),FJ(43)
      DIMENSION PBHAF(23,43),SOUN(23,43),CONT(23,43)
C
      COMMON/AA/IMAT,IMIN,ILES,IMAX,JMAT,JMIN,JLES,JMAX
      COMMON/CC/BC4,BC6,ERRS,ITEST,JTEST,NSTOP,NCOUNT,ERROR1
      COMMON/EE/PP,PTIME,TAU
      COMMON/FF/GOLD,G
      COMMON/PP/P,D,SOLD,PLP2,SUP2
      COMMON/JJ/AI,BI1,BI2,CI,DI,EI,FI,AJ,BJI,EJ2,CJ,DJ,EJ,FJ,CC41,CC42
      COMMON/KK/SOUN,CONT
      COMMON/CC/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAI,BETAK,DCHI
      COMMON/RR/NTRITE,ITMAX,NPRITE,NPLU,NPROB,NR,NL,NT,NITER,NOUT
C
    1 FORMAT(1H0,T5,'PP ARRAY,  I=2,ILES    J=1,JLES    NT='I4,'  NITER='I5,-
     Z      '  TAU='F10.6)
    2 FORMAT(1P10D13.5)
    3 FORMAT(1H0,T5,'PP('I2,','I2,')='1PD13.5,'  ERRS='D13.5)
    4 FORMAT(1HC,T5,'PBHAF ARRAY,  I=2,IMIN    J=2,JMIN    NT='I4,'  NITER='I5,-
     Z      '  TAU='F10.6)
C
C     -----PRESSURE ITERATION-----
  100 NITER=0
      ERROR1=ERROR
      NTEST=0
      NSTOP=0
      III=2
      JJJ=2
  101 BI=BI1
      BJ=BJ1
      CC42=CC41
      NTERU=0
  110 ITEST=III
      JTEST=JJJ
  200 DO 203 J=1,JLES
  201 DO 202 I=2,ILES
      PP(I,J)=PP(I,J)
  202 CONTINUE
  203 CONTINUE
C     -----FIRST HALF CHI STEP-----
  210 DO 229 J=2,JMIN
```

```
      JP=J+1
      JM=J-1
  211 DO 219 I=2,IMIN
      IM=I-1
      P=PP(I,J)
  212 CI(I)=-CC42*(PP(I,JP)+PP(I,JM)-P-P)-(CC43-1.0/SOUN(I,J))*P-G(I,J)
      IF (I.NE.IMIN) GO TO 214
  213 CI(IMIN)=CI(IMIN)-CI(IMIN)*BC6*B(ILES,J)
      GO TO 218
  214 IF (I.NE.2) GO TO 217
  215 EI(2)=CI(2)/BI
      FI(2)=DI(2)/BI
      GO TO 219
  217 EI(I)=CI(I)/(BI-AI(I)*EI(IM))
      FI(I)=(DI(I)-AI(I)*FI(IM))/(BI-AI(I)*EI(IM))
      GO TO 219
  218 EI(IMIN)=0.
      FI(IMIN)=(CI(IMIN)-AI(IMIN)*FI(IMNI))/(BI+CI(IMIN)-AI(IMIN)*EI(IMNI))
  219 CONTINUE
  220 DO 224 K=2,IMIN
      I=IMAX-K
  221 IF (K.NE.2) GO TO 223
  222 PBHAF(IMIN,J)=FI(IMIN)
      GO TO 224
  223 PBHAF(I,J)=FI(I)-EI(I)*PBHAF(I+1,J)
  224 CONTINUE
  229 CONTINUE
C     -----SECOND HALF CHI STEP-----
  230 DO 249 I=2,IMIN
  231 DO 237 J=2,JMIN
      JP=J+1
      JM=J-1
  232 P=PP(I,J)
      PHAF=PBHAF(I,J)
  233 CJ(J)=CC42*(PP(I,JP)+PP(I,JM)-P-P)+CC43*(P-PHAF-PHAF)
      IF (J.NE.2) GO TO 235
  234 CJ(2)=CJ(2)+AJ*C(I,2)/BC4
      EJ(2)=CJ/(AJ+BJ)
      FJ(2)=DJ(2)/(AJ+BJ)
      GO TO 237
  235 IF (J.EQ.JMIN) GO TO 236
      EJ(J)=CJ/(BJ-AJ*EJ(JM))
      FJ(J)=(CJ(J)-AJ*FJ(JM))/(BJ-AJ*EJ(JM))
      GO TO 237
  236 EJ(JMIN)=0.
      FJ(JMIN)=(CJ(JMIN)-AJ*FJ(JMNI))/(BJ-CJ-AJ*EJ(JMNI))
  237 CONTINUE
  240 DO 244 K=2,JMIN
      J=JMAX-K
  241 IF (K.NE.2) GO TO 243
  242 PB(I,JMIN)=FJ(JMIN)
      GO TO 244
  243 PB(I,J)=FJ(J)-EJ(J)*PB(I,J+1)
  244 CONTINUE
```

```
  249 CONTINUE
  115 DO 116 I=2,IMIN
      PB(I,1)=PB(I,2)-C(I,2)/BC4
      PB(I,JLES)=-PB(I,JMIN)
C 116 CONTINUE
  117 DO 118 J=2,JLES
      PB(ILES,J)=PB(IMIN,J)+BC6*B(ILES,J)
  118 CONTINUE
  119 NITER=NITER+1
  120 IF (NTEST.EQ.1) GO TO 140
C     -----TEST ERROR ARRAY FOR A FAILIURE-----
  121 DO 128 J=JTEST,JMIN
  122 DO 127 I=2,IMIN
      PABS=DABS(PB(I,J))+DABS(PP(I,J))
  123 IF (PABS.LT.1.D-10) GO TO 127
  124 ERRS=(PB(I,J)-PP(I,J))/PABS
      IF (DABS(ERRS).LT.ERROR1) GO TO 127
  125 NTEST=1
      III=I
      JJJ=J
      IF (NITER.LT.ITMAX) GO TO 131
 1250 NSTOP=1
      RETURN
  127 CONTINUE
  128 CONTINUE
  129 IF (ITEST.EQ.2.AND.JTEST.EQ.2) GO TO 132
  130 ITEST=2
      JTEST=2
      GO TO 121
  131 ITEST=III
      JTEST=JJJ
      GO TO 150
  132 IF (NTHRU.EQ.1) RETURN
  133 BI=BI2
      BJ=BJ2
      CC43=5.*CC41
      NTHRU=1
      ERRCR1=ERRCR/4.
      GO TO 150
C     -----TEST ERROR AT ONE POINT FOR FAILIURE-----
  140 PABS=DABS(PB(ITEST,JTEST))+DABS(PP(ITEST,JTEST))
      IF (PABS.GT.1.D-10) GO TO 142
  141 NTEST=0
      GO TO 121
  142 ERRS=(PB(ITEST,JTEST)-PP(ITEST,JTEST))/PABS
      IF (ERRS.LT.ERRCR1) GO TO 121
C     -----TEST WRITECUT-----
  150 IF (NITER.GE.ITMAX) GO TO 1560
      IF (MOD(NCOUNT,NTRITE).EQ.0.AND.MOD(NITER,MPRITE).EQ.0) GO TO 151
      GO TO 200
  151 WRITE (6,1) NT,NITER,TAU
      DO 152 J=1,JLES
      WRITE (6,2) (PB(I,J),I=2,ILES)
  152 CONTINUE
```

```
  155 WRITE (6,3) ITEST,JTEST,PB(ITEST,JTEST),ERRS
  156 IF (NITER.LT.ITMAX) GO TO 200
 1560 NSTOP=1
      RETURN
      END
```

```
C MASS2  - WIEBER SUBROUTINE MASS FOR SOLVING MULTICOMPONENT MASS TRANSFER
C          FLOWS WITH FMIX5. DOES BINARY TYPE TRANSFER OF TRACERS IN A
C          SOLVENT - COMPUTES ALL VALUES DESCRIBING SPECIE QUANTITIES.
C          CONTAINS ERROR CORRECTIONS FOR FINITE DIFFERENCING AND A
C          SCHEME FOR ELIMINATING NEGATIVE SPECIE DENSITIES. ADI IMPLICIT.
C
C
      SUBROUTINE MASS
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION PR(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
     Z  EYE(23,43),ENG(23,43)
      DIMENSION CONC(23,43),GAMMY(23,43),T(23,43),SUM(23,43)
      DIMENSION RAT1(23),RAT2(23)
      DIMENSION CONC1(23,43),CONC2(23,43),CONC3(23,43),CONC4(23,43),-
     Z  CONC5(23,43),SIZEX(6)
      DIMENSION SIGK(23,43,5),DIFFUZ(23,43,4),SOLD(23,43),SKUP(23,5)
      DIMENSION TERMU(23,43),TERMV(23,43),SKOLD(23,43),SKHAF(23,43)
      DIMENSION A1(23,43),B1(23,43),C1(23,43),D1(23,43),E1(23,43)
      DIMENSION F1(23,43),B2(23,43),E2(23,43),S(23,43),SIGKUP(23,5)
      DIMENSION DI(23),EI(23),FI(23),DJ(43),EJ(43),FJ(43)
      DIMENSION COEF1(23,43),COEF2(23,43),COEF3(23,43),COEF4(23,43)
C
      REAL*8 MX,MUSTAR,MOLWT(5),MOLMIX(23,43)
      REAL*8 NSC(23,43,4),MOLFR(23,43,5)
      REAL*8 NRE(23,43),NPR(23,43),MLES(23,43,4)
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/CC/PR,SIG,UR,VZ,EYE,ENG
      COMMON/NN/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/OO/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAT,BETAK,DCHI
      COMMON/QQ/SKCOEF,CC15,CC17,CC32,CC33,CC34,CC35,CC36,CC37
      COMMON/WW/RAT1,RAT2
      COMMON/XX/RIN,VIN,VOUT,ATURE,NANN
      COMMON/AC/NRE,NPR,GAMMY,MOLMIX,CONC
      COMMON/AD/NSC
      COMMON/AE/CC2,CC8,CC9,CC16,NPROP,NSPECY,NSPEC,MULTID,NY,NIN,NREACT
      COMMON/AF/SIGK,MOLFR
      COMMON/AG/CONC1,CONC2,CONC3,CONC4,CONC5
      COMMON/AH/SIGKUP,SKUP
      COMMON/AK/CC26,CC27
      COMMON/AL/MLES,T,GAMIN,GAMCW,TEMPCW,SIZEX
      COMMON/AC/EYECW,ENGCW,SIGCW,TPOW
C
C----------CALCULATE CORRECTION-----
  100 IF (BETAK.EQ.0.) GO TO 110
  101 DO 106 J=2,JMIN
      JP=J+1
  102 DO 105 I=2,IMIN
      IP=I+1
      XI=I+I-3
      TERMU(I,J)=CC15*(UR(IP,J)-UR(I,J)+(UR(IP,J)+UR(I,J))/XI)-
     Z  +CC26*(UR(IP,J)+UR(I,J))**2
```

```
      TERMV(I,J)=CC16*(VZ(I,JP)-VZ(I,J))+CC27*(VZ(I,JP)+VZ(I,J))**2
  103 IF (BETAK.GT.1.) GO TO 104
      TERMU(I,J)=BETAK*TERML(I,J)
      TERMV(I,J)=BETAK*TERMV(I,J)
      GO TO 105
  104 TERMU(I,J)=BKCOEF*DABS(TERMU(I,J))
      TERMV(I,J)=BKCOEF*DABS(TERMV(I,J))
  105 CONTINUE
  106 CONTINUE
C----------CALCULATE ACI AND S(I,J) COEFFICIENTS-----
  110 DO 116 J=2,JMIN
      JP=J+1
      JM=J-1
  111 DO 115 I=2,IMIN
      IP=I+1
      IM=I-1
      R1=RAT1(I)
      R2=RAT2(I)
      V=VZ(I,J)
      VJP=VZ(I,JP)
  112 A1(I,J)=-R2*UR(I,J)
      C1(I,J)=R1*UR(IP,J)
      B1(I,J)=C1(I,J)+A1(I,J)+CC32
      D1(I,J)=CC2*V
      F1(I,J)=CC2*VJP
      E1(I,J)=F1(I,J)-D1(I,J)-CC32
      B2(I,J)=V-VJP-CC33
      E2(I,J)=V-VJP+CC33
  113 XC=CONC(I,J)
      XM=MOLMIX(I,J)
      XN=NRE(I,J)
  114 COEF1(I,J)=CC36*R1*(CONC(IP,J)+XC)/(MOLMIX(IP,J)+XM)/(NRE(IP,J)+XN)
      COEF2(I,J)=CC36*R2*(XC+CONC(IM,J))/(XM+MOLMIX(IM,J))/(XN+NRE(IM,J))
      COEF3(I,J)=CC37*(CONC(I,JP)+XC)/(MOLMIX(I,JP)+XM)/(NRE(I,JP)+XN)
      COEF4(I,J)=CC37*(XC+CONC(I,JM))/(XM+MOLMIX(I,JM))/(XN+NRE(I,JM))
  115 CONTINUE
  116 CONTINUE
C----------CALC DIFFUSION TERMS-----
  120 DO 128 K=1,4
      MX=MOLWT(K)
  121 DO 127 J=2,JMIN
      JP=J+1
      JM=J-1
  122 DO 126 I=2,IMIN
      IP=I+1
      IM=I-1
  123 XF=MOLFR(I,J,K)
      XM=MLES(I,J,K)
      XN=NSC(I,J,K)
  124 DIFFUZ(I,J,K)=MX*(COEF1(I,J)*(MLES(IP,J,K)+XM)*(MOLFR(IP,J,K)-
     Z    -XF)/(NSC(IP,J,K)+XN)-COEF2(I,J)*(XM+MLES(IM,J,K))*(XF-MOLFR-
     Z    (IM,J,K))/(XN+NSC(IM,J,K))+COEF3(I,J)*(MLES(I,JP,K)+XM)*(MOLFR-
     Z    (I,JP,K)-XF)/(NSC(I,JP,K)+XN)-COEF4(I,J)*(XM+MLES(I,JM,K))*(XF-
     Z    -MOLFR(I,JM,K))/(XN+NSC(I,JM,K)))
```

```
  126 CONTINUE
  127 CONTINUE
  128 CONTINUE
C----------UPDATE INPUT MASS FLUX-----
  130 DO 135 K=1,5
  131 DO 132 I=2,NTUBE
      SKUP(I,K)=SIGKUP(I,K)*.5*(SIG(I,1)+SIG(I,2))
  132 CONTINUE
  133 DO 134 I=NANN,IMIN
      SKUP(I,K)=SIGKUP(I,K)*.5*(SIG(I,1)+SIG(I,2))/SIGOM
  134 CONTINUE
  135 CONTINUE
C----------ENTER MAIN LOOP-----
  200 DO 295 IJK=1,NIN
      DO 280 K=1,4
C----------SET UP OLD VALUE ARRAY-----
  201 DO 204 J=1,JLES
  202 DO 203 I=1,ILES
      SKOLD(I,J)=SIGK(I,J,K)
  203 CONTINUE
  204 CONTINUE
C----------CALCULATE EXPLICIT TERM S(I,J)-----
  210 DO 216 J=2,JMIN
      JP=J+1
      JM=J-1
  211 DO 215 I=2,IMIN
      IP=I+1
      IM=I-1
      R1=RAT1(I)
      R2=RAT2(I)
  212 XS=SKOLD(I,J)
  213 PART1=-CC35*(R1*(SKOLD(IP,J)+XS)*UR(IP,J)-R2*(XS+SKOLD(IM,J))-
Z     *UR(I,J)+CC2*((SKOLD(I,JP)+XS)*VZ(I,JP)-(XS+SKOLD(I,JM))*VZ(I,J)))
      PART4=0.
      IF (BETAK.EQ.0.) GO TO 214
      PART4=TERMU(I,J)*(SKOLD(IP,J)+SKOLD(IM,J)-XS-XS)+TERMV(I,J)-
Z     *(SKOLD(I,JP)+SKOLD(I,JM)-XS-XS)
  214 S(I,J)=CC17*(PART1+DIFFUZ(I,J,K)+PART4)
  215 CONTINUE
  216 CONTINUE
C----------FIRST HALF DTAU STEP------
  220 DO 227 J=2,JMIN
      JP=J+1
      JM=J-1
  221 DO 224 I=2,IMIN
      IM=I-1
  222 DI(I)=S(I,J)+D1(I,J)*SKOLD(I,JM)-E1(I,J)*SKOLD(I,J)-F1(I,J)-
Z     *SKOLD(I,JP)
      DENOM=B1(I,J)-A1(I,J)*EI(IM)
  223 EI(I)=C1(I,J)/DENOM
      FI(I)=(DI(I)-A1(I,J)*FI(IM))/DENOM
  224 CONTINUE
      SKHAF(IMIN,J)=FI(IMIN)
  225 DO 226 N=3,IMIN
```

```
      I=IMAX-N
      SKHAF(I,J)=FJ(I)-EJ(I)*SKHAF(I+1,J)
  226 CONTINUE
  227 CONTINUE
C----------SECOND HALF DTAU STEP-----
  230 DO 238 I=2,IMIN
      DJ(2)=VZ(I,2)*SKOLD(I,1)+B2(I,2)*SKOLD(I,2)-VZ(I,3)*SKOLD(I,3)-
Z     -CC34*SKHAF(I,2)
      EJ(2)=-VZ(I,3)/(B2(I,2)-VZ(I,2))
      FJ(2)=(DJ(2)-2.*VZ(I,2)*SKUP(I,K))/(B2(I,2)-VZ(I,2))
  231 DO 233 J=3,JMAX
      JP=J+1
      JM=J-1
  232 DJ(J)=VZ(I,J)*SKOLD(I,JM)+B2(I,J)*SKOLD(I,J)-VZ(I,JP)-
Z     *SKOLD(I,JP)-CC34*SKHAF(I,J)
      DENOM=B2(I,J)-VZ(I,J)*EJ(JM)
      EJ(J)=-VZ(I,JP)/DENOM
      FJ(J)=(DJ(J)-VZ(I,J)*FJ(JM))/DENOM
  233 CONTINUE
      DJ(JMIN)=VZ(I,JMIN)*SKOLD(I,JMAX)+B2(I,JMIN)*SKOLD(I,JMIN)-
Z     -VZ(I,JLES)*SKOLD(I,JLES)-CC34*SKHAF(I,JMIN)
      EJ(JMIN)=0.
      FJ(JMIN)=(DJ(JMIN)-(VZ(I,JMIN)+VZ(I,JLES))*FJ(JMAX))/(B2(I,JMIN)-
Z     -2.*VZ(I,JLES)-(VZ(I,JMIN)+VZ(I,JLES))*EJ(JMAX))
  235 SIGK(I,JMIN,K)=FJ(JMIN)
  236 DO 237 K=2,JMIN
      J=JMAX-K
      SIGK(I,J,K)=FJ(J)-EJ(J)*SIGK(I,J+1,K)
  237 CONTINUE
  238 CONTINUE
  250 DO 253 J=2,JMIN
  251 DO 252 I=2,IMIN
      IF (SIGK(I,J,K).LT.0.) SIGK(I,J,K)=0.
  252 CONTINUE
  253 CONTINUE
  280 CONTINUE
  290 IF (NREACT.GT.0) CALL PROPB
      CALL BOUNF
  295 CONTINUE
C----------CALCULATE REMAINING QUANTITIES-----
  300 DO 301 J=2,JMIN
      DO 301 I=2,IMIN
      SUM(I,J)=0.
  301 CONTINUE
  302 DO 303 K=1,4
      DO 303 J=2,JMIN
      DO 303 I=2,IMIN
      SUM(I,J)=SUM(I,J)+SIGK(I,J,K)
  303 CONTINUE
  304 DO 305 J=2,JMIN
      DO 305 I=2,IMIN
      SIGK(I,J,NSPECY)=SIG(I,J)-SUM(I,J)
  305 CONTINUE
  306 DO 309 J=2,JMIN
```

```
      DO 309 I=2,IMIN
  307 CONC1(I,J)=SIGK(I,J,1)/MOLWT(1)
      CONC2(I,J)=SIGK(I,J,2)/MOLWT(2)
      CONC3(I,J)=SIGK(I,J,3)/MOLWT(3)
      CONC4(I,J)=SIGK(I,J,4)/MOLWT(4)
      CONC5(I,J)=SIGK(I,J,5)/MOLWT(5)
      CONC(I,J)=CONC1(I,J)+CONC2(I,J)+CONC3(I,J)+CONC4(I,J)+CONC5(I,J)
      MOLMIX(I,J)=SIG(I,J)/CONC(I,J)
  308 MOLFR(I,J,1)=CONC1(I,J)/CONC(I,J)
      MOLFR(I,J,2)=CONC2(I,J)/CONC(I,J)
      MOLFR(I,J,3)=CONC3(I,J)/CONC(I,J)
      MOLFR(I,J,4)=CONC4(I,J)/CONC(I,J)
      MOLFR(I,J,5)=CONC5(I,J)/CONC(I,J)
  309 CONTINUE
C----------CALC MOL WEIGHTS OF PARTIAL MIXTURES-----
  310 DO 311 K=1,4
      MX=MOLWT(K)
      DO 311 J=2,JMIN
      DO 311 I=2,IMIN
      MOLS(I,J,K)=(MOLMIX(I,J)-MX*MOLFR(I,J,K))/(1.-MOLFR(I,J,K))
  311 CONTINUE
  312 CALL BOUND
      RETURN
      END
```

```
C  RESETT - WIEBER PROGRAM TO RESET OPERATING PARAMETERS AND MONITOR
C           VARIABLES WHEN MAIN PROGRAM IS FORTRANNED WITH ISD=N
C
C
      SUBROUTINE RESET
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      REAL*4 RVAVG,SAVG
C
      DIMENSION PR(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
     Z  EYE(23,43),ENG(23,43)
      DIMENSION OOLD(23,43),O(23,43)
      DIMENSION B(23,43),D(23,43),SOLD(23,43)
      DIMENSION PB(23,43),OONOLD(23,43)
      DIMENSION SOUN(23,43),CONT(23,43)
      DIMENSION Q(23,43),ENEW(23,43)
      DIMENSION RAT1(23),RAT2(23),RAT3(23)
      DIMENSION BETANT(23,43),BETAVR(23,43),BETAVZ(23,43)
      DIMENSION GOLD(23,43),EYOLD(23,43),-
     Z  VZOLD(23,43),UROLD(23,43),EOLD(23,43)
      DIMENSION AI(23),CI(23),DI(23),EI(23),FI(23),OJ(43),EJ(43),FJ(43)
      DIMENSION SIGK(23,43,5),SKUP(23,5)
      DIMENSION CONC1(23,43),CONC2(23,43),CONC3(23,43),CONC4(23,43),-
     Z  CONC5(23,43)
      DIMENSION SIGKUP(23,5),SMITIN(4)
      DIMENSION RVAVG(23),SAVG(43,6),REYUP(23)
      DIMENSION T(23,43),SIZEX(6),SMITUP(23,4),GAMUP(23),SMITOW(4)
C
      REAL*8 L,LT,MOLWT(5),MUSTAR,LOR,LORSQ,SIZER(12)
      REAL*8 MSTAR,MOLMIX(23,43)
      REAL*8 NSC(23,43,4),MOLFR(23,43,5)
      REAL*8 NRE(23,43),NPR(23,43),GAMMY(23,43),CONC(23,43)
      REAL*8 MLES(23,43,4),MOLFUP(23,5),MMIXUP(23),MLESUP(23,4)
      REAL*8 MMIXIN,MMIXOW,MLESIN(4),MLESOW(4)
C
      INTEGER DAY,ITERS(4000),SIZEI(15),SPECIE(6)
C
      LOGICAL ANS
C
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/BB/EYOLD,VZOLD,UROLD,EOLD
      COMMON/CC/PR,SIG,UR,VZ,EYE,ENG
      COMMON/DD/BC4,BC6,ERRS,ITEST,JTEST,NSTOP,NCOUNT,ERROR1
      COMMON/EE/PB,PTIME,TAU
      COMMON/FF/OOLD,O
      COMMON/GG/PUP,PUPMAX,SIGUP,SUPMAX,DSOIG,VELUP,VUPMAX,ROL,FREQ
      COMMON/HH/B,D,SOLD,PUP2,SUP2
      COMMON/II/NBUG,NRUN,MONTH,DAY
      COMMON/JJ/AI,BI1,BI2,CI,DI,EI,FI,AJ,BJ1,EJ2,CJ,DJ,EJ,FJ,CC41,CC42
      COMMON/KK/SOUN,CONT
      COMMON/LL/EYSTAR,RCUT,LT,EULNO,REYNO,STRONG,SMITNO,PRANNO,VREF
      COMMON/MM/Q,ENEW
```

```
      COMMON/AA/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/CC/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAI,BETAK,DCHI
      COMMON/CC/EKCOEF,CC15,CC17,CC32,CC33,CC34,CC35,CC36,CC37
      COMMON/RR/NTRITE,ITMAX,NPRITE,NFLU,NPROB,NR,NL,NT,NITER,NOUT
      COMMON/SS/TAUEND,NRITE,NTAPE,NSAVE,NFIRPE
      COMMON/TT/GOLD,BETAMT,BETAVR,BETAVZ,JSTART
      COMMON/UU/RAT1,RAT2
      COMMON/XX/RIN,VIN,VOUT,NTUBE,NANN
      COMMON/YY/DTOLD,NDTAU,NINOLD,NYOLD
      COMMON/AB/PSTAR,TSTAR,CSTAR,DVSTAR,CC3B,SPECIE
      COMMON/AC/NRE,NPR,GAMMY,MOLMIX,CONC
      COMMON/AD/NSC
      COMMON/AE/CC2,CC8,CC9,CC16,NPROP,NSPECY,NSPEC,MULTID,NY,NIN,NREACT
      COMMON/AF/SIGK,MOLER
      COMMON/AG/CONC1,CONC2,CONC3,CONC4,CONC5
      COMMON/AH/SIGKUP,SKUP
      COMMON/AI/RVAVG,SAVG
      COMMON/AK/CC26,CC27
      COMMON/AL/MLES,T,GAMIN,GAMOW,TEMPOW,SIZEX
      COMMON/AM/MOLFUP,MMIXUP,MLESUP,SMITUP,GAMUP,SMITIN,SMITOW,-
     Z    REYNIN,REYNOW,REYUP
      COMMON/AN/MMIXIN,MMIXOW,MLESIN,MLESOW
      COMMON/AO/EYEOW,ENGOW,SIGOW,TPOW
C
  100 IF (NDTAU.EQ.0) RETURN
      DTOLD=DTAU
      NINOLD=NIN
      NYOLD=NY
      PAUSE 'RESET DTAU, DCHI, NIN, NY, & NTRITE.'
      RETURN
      END
```

```
C WRITIT - WIEBER SUBROUTINE RITE FOR FMIX5 & 6 PROGRAMS. DISPLAYS EXTRA ARRAYS
C          CCLD, B, D, G, & SOUN IF NRUG=1 - CONTAINS WRITE ROUTINES
C          FOR LATER TAPE STORAGE. WRITES CONCENTRATION ARRAYS.
C
C
      SUBROUTINE RITE
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 VZSTOR,URSTOR,PRSTOR,SGSTOR,TPSTOR,TIME
      REAL*4 RVAVG,SAVG
      REAL*4 CASTOR,C1STOR,C2STOR,C3STOR,C4STOR,C5STOR
C
      DIMENSION PB(23,43),PR(23,43),SIG(23,43),UR(23,43),VZ(23,43),-
     Z  EYE(23,43),ENG(23,43)
      DIMENSION CCLD(23,43),G(23,43),CONC(23,43)
      DIMENSION B(23,43),D(23,43),SCLD(23,43)
      DIMENSION SOUN(23,43),CONT(23,43),EYEN(23,43),ENGN(23,43)
      DIMENSION VZSTOR(23,43,100),URSTOR(23,43,100),PRSTOR(23,43,100),-
     Z  SGSTOR(23,43,100),TPSTOR(23,43,100)
      DIMENSION T(23,43),TIME(100),NTT(100)
      DIMENSION CONC1(23,43),CONC2(23,43),CONC3(23,43),CONC4(23,43),-
     Z  CONC5(23,43)
      DIMENSION CASTOR(23,43,100),C1STOR(23,43,100),C2STOR(23,43,100),-
     Z  C3STOR(23,43,100),C4STOR(23,43,100),C5STOR(23,43,100)
      DIMENSION CON(23,43),RVAVG(23),SAVG(43,6)
C
      REAL*8 MUSTAR,PSTAR,LT,MOLWT(5),MOLMIX(23,43),MLES(23,43,4),SIZEX(6)
      REAL*8 NRE(23,43),NPR(23,43),GAMMY(23,43)
C
      INTEGER DAY,SPECIE(6)
C     INTEGER NAME(14)/4H   C,4H   B,4H   D,4H   G,4HSOUN,4H  PB,4HTEMP,
C    14H  PR,4H SIG,4H  UR,4H  VZ,4HEYEN,4HENGN,4HCONT/
CC
      DATA LOW1,LOW2/1,2/
C
      COMMON/AA/IMAT,IMIN,ILES,IMAX,JMNT,JMIN,JLES,JMAX
      COMMON/CC/PR,SIG,UR,VZ,EYE,ENG
      COMMON/EE/PB,PTIME,TAU
      COMMON/FF/CCLD,G
      COMMON/GG/PUP,PUPMAX,SIGUP,SUPMAX,DSQIG,VELUF,VUPMAX,RDL,FREQ
      COMMON/HH/B,D,SCLD,PUFF,SUP2
      COMMON/II/NBUG,NRUN,MONTH,DAY
      COMMON/KK/SOUN,CONT
      COMMON/LL/EYSTAR,ROUT,LT,EULNO,REYNO,STRONO,SMITNO,PRANNO,VREF
      COMMON/NN/PSTAR,SGSTAR,MUSTAR,MOLWT,INCOMP
      COMMON/OO/DTAU,TCYC,ERROR,PHI,THETA,PSI,BETAM,BETAV,BETAI,BETAK,DCHI
      COMMON/RR/NTRITE,ITMAX,NPRITE,NFLU,NPROB,NP,NL,NT,NITER,NOUT
      COMMON/SS/IAUENC,NRITE,NTAPE,NSAVE,NFIRPE
      COMMON/XX/RIN,VIN,VOUT,NTUBE,NANN
      COMMON/AB/MSTAR,TSTAR,CSTAR,CVSTAR,CC3R,SPECIE
      COMMON/AC/NRE,NPR,GAMMY,MOLMIX,CONC
      COMMON/AE/CC2,CC8,CC9,CC16,NPROP,NSPECY,NSPEC,MULTID,NY,NIN,NREACT
      COMMON/AG/CONC1,CONC2,CONC3,CONC4,CONC5
      COMMON/AI/RVAVG,SAVG
```

```
      COMMON/AL/MLES,T,GAMIN,GAMOW,TEMPOW,SIZEX
      COMMON/AO/EYEOW,ENGOW,SIGOW,TFOW
C
    2 FORMAT(1HC,T2,'RUN #'I2,' ON 'I2,'/'I2,'/72')
    3 FORMAT(2HC ,'INCREMENT SIZES    DIMENSIONS    DIMENSIONLESS # S',-
Z     7X,'VELOCITIES',14X,'P, RHO, T',11X,'REFERENCES',7X,'CONTROLS'/-
Z     T2,'----------------  ------------  ----------------',-
Z     ' ---------------------  --------------------  ----------------',-
Z     ' ----------')
    4 FORMAT(T2,'DTAU  ='1PD10.4,'  RIN   ='0PF6.2,'  EULNO ='1PD10.4,-
Z     '  VELUP='D10.4,' MEAN  PUP  ='D10.4,' MEAN  PSTAR ='D10.4,-
Z     '  FHI  ='0PF4.2)
    5 FORMAT(T2,'DCHI  ='1PD10.4,'  RCUT  ='0PF6.2,'  STRONG='1PD10.4,-
Z     '       ='D10.4,' MAX        ='D10.4,' MAX   SGSTAR='D10.4,-
Z     '  THETA='0PF4.2)
    6 FORMAT(T2,'ERROR ='1PD10.4,'  LENGTH='0PF6.2,'  REYNO ='1PD10.4,-
Z     '  VIN  ='D10.4,'       SIGUP='D10.4,' MEAN  TSTAR ='D10.4,-
Z     '  FSI  ='0PF4.2)
    7 FORMAT(T2,'DSOIG ='1PD10.4,'  RCL   ='0PF6.4,'  PRANNO='1PD10.4,-
Z     '  VOUT ='D10.4,'            ='D10.4,' MAX   MSTAR ='D10.4,-
Z     '  BETAM='0PF4.2)
    8 FORMAT(T2,'DETA  ='1PD10.4,'  TCYC  ='0PF6.4,'  SMITNO='1PD10.4,-
Z     '  VREF ='D10.4,'       TUP  ='D10.4,' MEAN  CSTAR ='D10.4,-
Z     '  BETAV='0PF4.2)
    9 FORMAT(T2,'TAUEND='1PD10.4,'  FREQ  ='0PF6.2,49X,'='1PD10.4,-
Z     ' MAX   EYSTAR='D10.4,'  BETAK='0PF4.2/T101,'EYY   ='1PD10.4)
C
   10 FORMAT(1H+,T25,'INCOMPRESSIBLE STARTUP OF LAMINAR FLOW OF BLOOD-
Z IN AN INFINITE TUBE WITH NO-SLIP WALLS,'/T25,'NO INITIAL PRESSURE-
Z GRADIENT IS SPECIFIED.   PROBLEM #1'/)
   11 FORMAT(1H+,T25,'INCOMPRESSIBLE STARTUP OF LAMINAR FLOW OF BLOOD-
Z IN AN INFINITE TUBE WITH NO-SLIP WALLS,'/T25,'A LINEAR INITIAL-
Z PRESSURE GRADIENT IS SPECIFIED.   PROBLEM #2'/)
   12 FORMAT(1H+,T25,'REMOVAL OF A DIAPHRAGM DIVIDING AN UPSTREAM HIGH-
Z PRESSURE RESERVOIR OF AN IDEAL GAS AND A DOWNSTREAM'/T25,'LOW-
Z PRESSURE RESERVOIR OF THE SAME GAS IN A CLOSED-END SHOCK TUBE.   PROBLEM #3'/)
   13 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH A COSINE-
Z CYCLIC + MEAN UPSTREAM PRESSURE,'/T25,'FLOW STARTS AT CYCLIC VALUES.   PROBLEM #4'/)
   14 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH A COSINE-
Z CYCLIC + MEAN UPSTREAM PRESSURE,'/T25,'FLOW STARTS AT REST.   PROBLEM #5'/)
   15 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH CONSTANT-
Z UNIFORM INPUT AND INITIALLY UNIFORM DOWN THE TUBE.'/)
   16 FORMAT(1H+,T25,'SUDDEN STOPPING OF AN INITIAL PLUG FLOW IN A TUBE WHOSE-
Z ENTRANCE AND EXIT IS ABRUPTLY BLOCKED.'/T25,'USED AS A TEST OF THE PRESSURE-
Z FIELD RELAXATION.   PROBLEM #7'/)
   17 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH A CONSTANT COAXIAL-
Z ENTRY AND INITIALLY UNIFORMLY COAXIAL DOWN THE TUBE.'/T25,'BOTH FLUIDS ARE-
Z INCOMPRESSIBLE AND IDENTICAL EXCEPT FOR VELOCITIES.   PROBLEM #8'/)
   18 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH CONSTANT-
Z COAXIAL ENTRY AND INITIALLY ZERO FLOW DOWN THE TUBE.'/T25,'BOTH-
Z FLUIDS ARE IDENTICAL EXCEPT FOR VELOCITIES. NO-SLIP WALLS. PROBLEM #9'/)
   19 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH CONSTANT-
Z COAXIAL ENTRY AND INITIALLY ZERO FLOW DOWN THE TUBE.'/T25,'BOTH-
Z FLUIDS ARE IDENTICAL EXCEPT FOR VELOCITIES. FREE-SLIP WALLS. PROBLEM #10'/)
```

```
   20 FORMAT(1H+,T25,'INCOMPRESSIBLE LAMINAR TUBE FLOW WITH CONSTANT-
Z PARABOLIC COAXIAL ENTRY AND INITIALLY PARABOLIC EXIT VELOCITY.'/T25,-
Z' FLUIDS ARE H2 & N2 IN THE CORE AND I2 & N2 IN THE ANNULUS INITIALLY.--
Z CHURCHILL PROBLEM #11'/)
   21 FORMAT(1H+,T25,'COMPRESSIBLE LAMINAR TUBE FLOW WITH CONSTANT-
Z COAXIAL ENTRY, IDENTICAL FLUIDS,'/T25,'AND FLUID INITIALLY AT REST.-
Z PROBLEM #12')
   22 FORMAT(1H+,T25,'COMPRESSIBLE LAMINAR TUBE FLOW WITH CONSTANT-
Z COAXIAL MASS FLOW ENTRY, FLUID INITIALLY AT REST.'/T25,'CENTER FLUID-
Z IS AIR AT 70 DEG F, OUTER FLUID HAS CO, C2, CC2, H2O, AND N2, AND IS-
Z HOT. PROBLEM #13'/)
   40 FORMAT(1HC,T5,'GRID',18X,'PROBLEM TYPE',20X,'LOOP CONTROLS',11X,-
Z     'WRITE CONTROLS'/T5,'-----    ------------------------------------
Z----    --------------------    --------------------')
   41 FORMAT(T5,'NR='I2,'    NPROB='I2,'  INCOMP='I1,'  NSPECY='I1,-
Z     '  MULTID='I1,'    MIN ='I2,'  NY   ='I2,7X,'NTRITE='I5,-
Z     '  NRITE='I1)
   42 FORMAT(T5,'NL='I2,'    NFLU ='I2,'  NPROP ='I1,'  NREACT='I1,-
Z     '  NFIRPB='I1,'    NOUT='I2,'  ITMAX='I5,'    NPRITE='I5,-
Z     '  NTAPE='I1)
   43 FORMAT(1HC,T11,'SPECIE',5X,A4,4(6X,A4))
   44 FORMAT(T11,'CONC IN   ',F6.4,4(4X,F6.4),10X,'TPOW ='F7.2,-
Z     4X,'EYECW='1PD10.4)
   45 FORMAT(T11,'CONC OUT  ',F6.4,4(4X,F6.4),10X,'SIGOW='F7.5,-
Z     4X,'ENGCW='1PD10.4)
   50 FORMAT(1HC,T6,A4,' ARRAY  I='I2,' TO 'I2,'  & J='I2,' TO 'I2,-
Z     '   NT='I4,' NITER='I4,' TAU='F10.3)
   51 FORMAT(1P10D13.5)
   52 FORMAT(1HC,T6,A4,' AXIAL AVERAGES  J='I2,' TO 'I2)
   53 FORMAT(1HO,T2,'TOTALS- ',6(A4,'=',1PD12.5,3X))
C
      ENTRY RITEA
C
C     -----WRITE INITIAL HEADINGS-----
  100 EYY=EYSTAR/(VREF*VREF)
      CETA=DSCIG/RDL
      TP=(TPOW+1.)*TSTAR
      WRITE (6,2) NRUN,MONTH,DAY
  102 GO TO (1020,1021,1022,1023,1024,1025,1026,1027,1028,1029,103,1030,1031),NPROB
 1020 WRITE (6,10)
      GO TO 104
 1021 WRITE (6,11)
      GO TO 104
 1022 WRITE (6,12)
      GO TO 104
 1023 WRITE (6,13)
      GO TO 104
 1024 WRITE (6,14)
      GO TO 104
 1025 WRITE (6,15)
      GO TO 104
 1026 WRITE (6,16)
      GO TO 104
 1027 WRITE (6,17)
```

```
      GC TC 104
 1C28 WRITE (6,18)
      GO TO 104
 1029 WRITE (6,19)
      GC TC 104
  103 WRITE (6,20)
      GC TC 104
 1030 WRITE (6,21)
      GO TO 104
 1031 WRITE (6,22)
  104 IF (NSPECY.EQ.1) GC TC 105
      C1IN=.5*(CONC1(2,1)+CONC1(2,2))
      C2IN=.5*(CONC2(2,1)+CONC2(2,2))
      C3IN=.5*(CONC3(2,1)+CONC3(2,2))
      C4IN=.5*(CONC4(2,1)+CONC4(2,2))
      C5IN=.5*(CONC5(2,1)+CONC5(2,2))
      C1OUT=.5*(CONC1(IMIN,1)+CONC1(IMIN,2))
      C2CUT=.5*(CONC2(IMIN,1)+CONC2(IMIN,2))
      C3CUT=.5*(CONC3(IMIN,1)+CONC3(IMIN,2))
      C4OUT=.5*(CONC4(IMIN,1)+CONC4(IMIN,2))
      C5CUT=.5*(CONC5(IMIN,1)+CONC5(IMIN,2))
  105 WRITE (6,3)
      WRITE (6,4) DTAU,RIN,EULNC,VELUP,PUP,PSTAR,PHI
      WRITE (6,5) DCHI,ROUT,STRONC,VUPMAX,PUPMAX,SGSTAR,THETA
      WRITE (6,6) ERRCR,LT,REYNC,VIN,SIGUP,TSTAR,PSI
      WRITE (6,7) DSQIG,RDL,PRANNO,VOLT,SUPMAX,MSTAR,BETAM
      WRITE (6,8) BETA,TCYC,SMITNO,VREF,TUP,CSTAR,BETAV
      WRITE (6,9) TAUEND,FREC,TUPMAX,EYSTAR,BETAK,EYY
      WRITE (6,40)
      WRITE (6,41) AR,NPRCB,INCOMP,NSPECY,MULTID,NIN,NY,NTRITE,NRITE
      WRITE (6,42) NL,NFLU,NPRCP,NREACT,NFIRER,NCUT,ITMAX,NPRITE,NTAPE
 1050 IF (NSPECY.EQ.1) GO TO 106
      WRITE (6,43) SPECIE
      WRITE (6,44) C1IN,C2IN,C3IN,C4IN,C5IN,TP,EYECW
      WRITE (6,45) C1OUT,C2OUT,C3OUT,C4OUT,C5OUT,SIGOW,ENGOW
  106 IF (NTAPE.EQ.0) GC TC 109
 1060 DO 1067 N=1,100
 1061 DC 1065 J=1,43
 1062 DO 1064 I=1,23
 1063 VZSTOR(I,J,N)=0.
      URSTOR(I,J,N)=0.
      PRSTOR(I,J,N)=0.
      SGSTOR(I,J,N)=0.
      TPSTOR(I,J,N)=0.
 1064 CONTINUE
 1065 CONTINUE
 1066 IF (NTT(N).EQ.0) GC TC 107
      NTT(N)=0
      TIME(N)=0.
 1067 CONTINUE
  107 WRITE (12) PUP,PUPMAX,RCUT,DTAU,EULNC,DCHI,SIGUP,SUPMAX,LT,DSQIG,-
Z REYNC,PHI,VELUP,VUPMAX,TCYC,RDL,ERRCR,THETA,VREF,FREQ,STRONO,BETAM,-
Z BETAV,BETAI,BETAK,SMITNC,VIN,VCUT,RIN,PSTAR,SGSTAR,EYSTAR,EYY,TAUEND,-
Z NTRITE,NTAPE,NL,ITMAX,NFLU,NPRITE,NPRCB,NREACT,NY,NIN,NSPECY,MULTID,-
```

```
      Z WRITE,AR,AREGF,SPECIE,NCUT,AFIRFB,NSAVE,INCOMP,MONTH,DAY,NRUN
      C
            ENTRY RITEF
      C
      C     -----WRITE DEBUG ARRAYS-----
            EYY=EYSTAR/(VREF*VREF)
        109 IF (NRITE.EQ.0) GO TO 500
        110 IF (NBUG.NE.1) GO TO 200
      C 111 WRITE (6,50) NAME(1),LOW2,ILES,LOW1,JLES,NT,NITER,TAU
      C 112 DO 113 J=1,JLES
      C     WRITE (6,51) (QOLD(I,J),I=2,ILES)
      C 113 CONTINUE
      C 114 WRITE (6,50) NAME(2),LOW2,ILES,LOW2,JLES,NT,NITER,TAU
      C 115 DO 116 J=2,JLES
      C     WRITE (6,51) (B(I,J),I=2,ILES)
      C 116 CONTINUE
      C 117 WRITE (6,50) NAME(3),LOW2,ILES,LOW2,JLES,NT,NITER,TAU
      C 118 DO 119 J=2,JLES
      C     WRITE (6,51) (D(I,J),I=2,ILES)
      C 119 CONTINUE
      C 120 WRITE (6,50) NAME(4),LOW2,IMIN,LOW2,JMIN,NT,NITER,TAU
      C 121 DO 122 J=2,JMIN
      C     WRITE (6,51) (G(I,J),I=2,IMIN)
      C 122 CONTINUE
            IF (INCOMP.EQ.1) GO TO 400
      C 123 WRITE (6,50) NAME(5),LOW2,IMIN,LOW2,JMIN,NT,NITER,TAU
      C 124 DO 125 J=2,JMIN
      C     WRITE (6,51) (SOUN(I,J),I=2,IMIN)
      C 125 CONTINUE
      C     -----MAIN WRITECUT-----
        200 CONTINUE
      C 201 WRITE (6,50) NAME(6),LOW2,ILES,LOW1,JLES,NT,NITER,TAU
      C 202 DO 203 J=1,JLES
      C     WRITE (6,51) (PP(I,J),I=2,ILES)
      C 203 CONTINUE
      C     -----TRANSFORM VARIABLES-----
        300 IF (INCOMP.EQ.1) GO TO 400
            DO 303 J=1,JLES
        301 DO 302 I=1,ILES
            EYEN(I,J)=EYE(I,J)/EYY
            ENGN(I,J)=ENG(I,J)/EYY
        302 CONTINUE
        303 CONTINUE
        400 IF (NRITE.EQ.0) GO TO 500
      C 401 WRITE (6,50) NAME(8),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
      C 402 DO 403 J=1,JLES
      C     WRITE (6,51) (PR(I,J),I=2,IMIN)
      C 403 CONTINUE
            IF (INCOMP.EQ.1) GO TO 407
      C 404 WRITE (6,50) NAME(9),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
      C 405 DO 406 J=1,JLES
      C     WRITE (6,51) (SIG(I,J),I=2,IMIN)
      C 406 CONTINUE
      C 407 WRITE (6,50) NAME(10),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
```

```
C  408 DO 409 J=1,JLES
C       WRITE (6,51) (UR(I,J),I=2,IMIN)
C  409 CONTINUE
C  410 WRITE (6,50) NAME(11),LOW2,IMIN,LOW1,JMAX,NT,NITER,TAU
C  411 DO 412 J=1,JMAX
C       WRITE (6,51) (VZ(I,J),I=2,IMIN)
C  412 CONTINUE
       IF (INCOMP.EQ.1) GO TO 422
C 4120 WRITE (6,50) NAME(13),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
C 4121 DO 4122 J=1,JLES
C       WRITE (6,51) (ENGN(I,J),I=2,IMIN)
C 4122 CONTINUE
C  413 WRITE (6,50) NAME(12),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
C  414 DO 415 J=1,JLES
C       WRITE (6,51) (EYEN(I,J),I=2,IMIN)
C  415 CONTINUE
   416 WRITE (6,50) NAME(7),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   417 DO 418 J=1,JLES
       WRITE (6,51) (T(I,J),I=2,IMIN)
   418 CONTINUE
C  422 WRITE (6,50) NAME(14),LOW2,IMIN,LOW2,JMIN,NT,NITER,TAU
C  423 DO 424 J=2,JMIN
C       WRITE (6,51) (CONT(I,J),I=2,IMIN)
C  424 CONTINUE
 C----------WRITE CONCENTRATION ARRAYS-----
   450 IF (NSPECY.EQ.1) GO TO 500
   451 WRITE (6,50) SPECIE(1),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   452 DO 453 J=1,JLES
       WRITE (6,51) (CONC1(I,J),I=2,IMIN)
   453 CONTINUE
   454 WRITE (6,50) SPECIE(2),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   455 DO 456 J=1,JLES
       WRITE (6,51) (CONC2(I,J),I=2,IMIN)
   456 CONTINUE
       IF (NSPECY.EQ.2) GO TO 466
   457 WRITE (6,50) SPECIE(3),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   458 DO 459 J=1,JLES
       WRITE (6,51) (CONC3(I,J),I=2,IMIN)
   459 CONTINUE
       IF (NSPECY.EQ.3) GO TO 466
   460 WRITE (6,50) SPECIE(4),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   461 DO 462 J=1,JLES
       WRITE (6,51) (CONC4(I,J),I=2,IMIN)
   462 CONTINUE
       IF (NSPECY.EQ.4) GO TO 466
   463 WRITE (6,50) SPECIE(5),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   464 DO 465 J=1,JLES
       WRITE (6,51) (CONC5(I,J),I=2,IMIN)
   465 CONTINUE
   466 WRITE (6,50) SPECIE(6),LOW2,IMIN,LOW1,JLES,NT,NITER,TAU
   467 DO 468 J=1,JLES
       WRITE (6,51) (CONC(I,J),I=2,IMIN)
   468 CONTINUE
 C----------COMPUTE CONCENTRATION AVERAGES-----
```

```
  470 IF (NSPECY.EQ.1) GO TO 500
  471 DO 481 J=2,JMIN
      JP=J+1
  472 DO 474 I=2,IMIN
      IP=I+1
  473 RVAVG(I)=.25*(I-1)*(VZ(I,J)+VZ(IP,J)+VZ(I,JP)+VZ(IP,JP))*DSCIG
  474 CONTINUE
      GO TO (500,478,477,476,475),NSPECY
  475 K=5
      CALL SIMPLE (CONC5,J,K)
  476 K=4
      CALL SIMPLE (CONC4,J,K)
  477 K=3
      CALL SIMPLE (CONC3,J,K)
  478 K=2
      CALL SIMPLE (CONC2,J,K)
  479 K=1
      CALL SIMPLE (CONC1,J,K)
  480 K=6
      CALL SIMPLE (CONC,J,K)
  481 CONTINUE
  482 SUM1=0.
      SUM2=0.
      SUM3=0.
      SUM4=0.
      SUM5=0.
      SUM6=0.
  483 DO 485 J=2,JMIN
  484 SUM1=SUM1+SAVG(J,1)
      SUM2=SUM2+SAVG(J,2)
      SUM3=SUM3+SAVG(J,3)
      SUM4=SUM4+SAVG(J,4)
      SUM5=SUM5+SAVG(J,5)
      SUM6=SUM6+SAVG(J,6)
  485 CONTINUE
  486 SUM1=SUM1/JMNI
      SUM2=SUM2/JMNI
      SUM3=SUM3/JMNI
      SUM4=SUM4/JMNI
      SUM5=SUM5/JMNI
      SUM6=SUM6/JMNI
C----------WRITE RADIAL AVERAGE CONCENTRATION ARRAYS-----
  490 WRITE (6,52) SPECIE(1),LOW2,JMIN
      WRITE (6,51) (SAVG(J,1),J=2,JMIN)
  491 WRITE (6,52) SPECIE(2),LOW2,JMIN
      WRITE (6,51) (SAVG(J,2),J=2,JMIN)
      IF (NSPECY.EQ.2) GO TO 495
  492 WRITE (6,52) SPECIE(3),LOW2,JMIN
      WRITE (6,51) (SAVG(J,3),J=2,JMIN)
      IF (NSPECY.EQ.3) GO TO 495
  493 WRITE (6,52) SPECIE(4),LOW2,JMIN
      WRITE (6,51) (SAVG(J,4),J=2,JMIN)
      IF (NSPECY.EQ.4) GO TO 495
  494 WRITE (6,52) SPECIE(5),LOW2,JMIN
```

```
      WRITE (6,51) (SAVG(J,5),J=2,JMIN)
  495 WRITE (6,52) SPECIE(6),LOW2,JMIN
      WRITE (6,51) (SAVG(J,6),J=2,JMIN)
  496 WRITE (6,53) SPECIE(1),SUM1,SPECIE(2),SUM2,SPECIE(3),SUM3,-
     Z     SPECIE(4),SUM4,SPECIE(5),SUM5,SPECIE(6),SUM6
     C----------STORE VALUES FOR TAPING-----
  500 IF (NTAPE.EQ.0) GO TO 600
      IF (NT.EQ.0) GO TO 600
  501 DO 507 J=1,JLES
  502 DO 506 I=1,ILES
  503 VZSTOR(I,J,NT)=VZ(I,J)
      URSTOR(I,J,NT)=UR(I,J)
      PRSTOR(I,J,NT)=PR(I,J)
  504 IF (INCOMP.EQ.1) GO TO 5050
  505 SGSTOR(I,J,NT)=SIG(I,J)
      TPSTOR(I,J,NT)=T(I,J)
 5050 IF (NSPECY.LT.2) GO TO 506
      CASTOR(I,J,NT)=CONC(I,J)
      C1STOR(I,J,NT)=CONC1(I,J)
      C2STOR(I,J,NT)=CONC2(I,J)
 5051 IF (NSPECY.EQ.2) GO TO 506
      C3STOR(I,J,NT)=CONC3(I,J)
 5052 IF (NSPECY.EQ.3) GO TO 506
      C4STOR(I,J,NT)=CONC4(I,J)
 5053 IF (NSPECY.EQ.4) GO TO 506
      C5STOR(I,J,NT)=CONC5(I,J)
  506 CONTINUE
  507 CONTINUE
      TIME(NT)=TAU
      NTT(NT)=NT
  510 WRITE (12) NTT(NT),TIME(NT)
  511 WRITE (12) ((VZSTOR(I,J,NT),I=1,ILES),J=1,JLES)
      WRITE (12) ((URSTOR(I,J,NT),I=1,ILES),J=1,JLES)
      WRITE (12) ((PRSTOR(I,J,NT),I=1,ILES),J=1,JLES)
  512 IF (INCOMP.EQ.1) GO TO 514
  513 WRITE (12) ((SGSTOR(I,J,NT),I=1,ILES),J=1,JLES)
      WRITE (12) ((TPSTOR(I,J,NT),I=1,ILES),J=1,JLES)
  514 IF (NSPECY.EQ.1) GO TO 600
      WRITE (12) ((CASTOR(I,J,NT),I=1,ILES),J=1,JLES)
      WRITE (12) ((C1STOR(I,J,NT),I=1,ILES),J=1,JLES)
      WRITE (12) ((C2STOR(I,J,NT),I=1,ILES),J=1,JLES)
  515 IF (NSPECY.EQ.2) GO TO 600
      WRITE (12) ((C3STOR(I,J,NT),I=1,ILES),J=1,JLES)
  516 IF (NSPECY.EQ.3) GO TO 600
      WRITE (12) ((C4STOR(I,J,NT),I=1,ILES),J=1,JLES)
  517 IF (NSPECY.EQ.4) GO TO 600
      WRITE (12) ((C5STOR(I,J,NT),I=1,ILES),J=1,JLES)
  600 RETURN
      END
```

```
C SIMPS - WIEBER SUBROUTINE SIMPLE FOR USING SIMPSONS 1/3RD AND 3/8THS
C         RULE TO CALCULATE RADIUS AVERAGE CONCENTRATION FLUXES
C
      SUBROUTINE SIMPLE (CCN,J,K)
C
      REAL*8 CCN
C
      DIMENSION RVAVG(23),POINT(23),CCN(23,43),SAVG(43,6)
C
      COMMON/AA/IMNI,IMIN,ILES,IMAX,JMNI,JMIN,JLES,JMAX
      COMMON/AI/RVAVG,SAVG
C
  100 DO 101 I=2,IMIN
      POINT(I)=RVAVG(I)*(CCN(I,J)+CCN(I+1,J))
  101 CONTINUE
      SUM=0.
  102 IF (MOD(ILES,2).NE.0) GO TO 103
      IFIRST=2
      GO TO 104
  103 SUM=.375*(3.*POINT(2)+3.*POINT(3)+POINT(4))
      IFIRST=5
  104 DO 105 I=IFIRST,IMNI,2
      S=(POINT(I-1)+4.*POINT(I)+POINT(I+1))/3.
      SUM=SUM+S
  105 CONTINUE
  106 SAVG(J,K)=SUM/IMNI
      RETURN
      END
```

BIBLIOGRAPHY

1. Akima, H., "A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures", J. Assoc. Comp. Mach., 17, 589-602 (1970).

2. Amsden, A. A., "The Particle-in-Cell Method for the Calculation of the Dynamics of Compressible Fluids", Los Alamos Scientific Laboratory Report No. LA-3466 (1966), 170 pages.

3. Amsden, A. A., and F. H. Harlow, "The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flows", Los Alamos Scientific Laboratory Report No. LA 4370, 85 pages (1970).

4. Bird, R. B., W. E. Stewart, and E. N. Lightfoot, Transport Phenomena, Wiley & Sons, New York (1960).

5. Brailovskaya, I. Yu., "A Difference Scheme for Numerical Solution of the Two-Dimensional, Nonstationary Navier-Stokes Equations for a Compressible Gas", Soviet Physics-Doklady, 10, 107-110 (1965).

6. Brian, P. L. T., "A Finite-Difference Method of High-Order Accuracy for the Solution of Three Dimensional Transient Heat Conduction Problems", AIChE J., 7, 367-370 (1961).

7. Brokaw, R. S., "Predicting Transport Properties of Dilute Gases", I&EC Process Design and Development, 8, 240-253 (1969).

8. Brokaw, R. S. and D. A. Bittker, "Carbon Monoxide Oxidation Rates Computed for Automobile Exhaust Manifold Reactor Conditions", NASA TN D-7024 (1970), 19 pages.

9. Burstein, S. Z., "Finite Difference Calculations for Hydrodynamic Flows Containing Discontinuities", J. Comp. Phys., 2, 198-222 (1967).

10. Canright, R. B., Jr., and P. Swigert, "Plot3D-A Package of Fortran Subprograms to Draw Three-Dimensional Surfaces", NASA TM-X-1598 (1968), 32 pages.

11. Cheng, S. I., "Numerical Integration of Navier-Stokes Equations", AIAA 8th Aerospace Sci. Mtg., New York, N.Y., Jan. 19-21, 1970, paper no. 70-2, 14 pages. See also AIAA J., 8, 2115-2122 (1970).

12. Courant, R., K. Friedrichs, and H. Lewy, "On the Partial Difference Equations of Mathematical Physics", IBM Journal, 11, 215-234 (1967). Originally published in German in Mathematische Annalen, 100, 32-74 (1928).

13. Daly, B. J., "Numerical Study of Two Fluid Rayleigh-Taylor Instability", Phys. Fluids, 10, 297-307 (1967).

14. Donovan, L. F., "Numerical Solution of the Unsteady Navier-Stokes Equations and Application to Flow in a Rectangular Cavity with a Moving Wall", NASA Tech. Note TN D-6312, 56 pages (1971).

15. Edelman, R. B., O. Fortune, and G. Weilerstein, "Some Observations on Flows Described by Coupled Mixing and Kinetics", in Emissions from Continuous Combustion Systems, W. Cornelius and W. G. Agnew, editors, Plenum Press, New York & London (1972), pp. 55-87 plus discussion.

16. Evans, M. W., and F. H. Harlow, "Calculation of Unsteady Supersonic Flow Past a Circular Cylinder", ARS Journal, 29, 46-48 (1959).

17. Fernandez, R. C., "Pulsatile Flow through a Bifurcation with Applications to Arterial Disease", PH.D. thesis, Dept. of Chemical Engineering, University of Toledo, 118 pages (1972).

18. Forsythe, G. E., and W. R. Wasow, Finite-Difference Methods for Partial Differential Equations, Wiley and Sons, Inc., New York (1960).

19. Friedmann, M., J. Gillis, and N. Liron, "Laminar Flow in a Pipe at Low and Moderate Reynolds Numbers", Appl. Sci. Res., 19, 426-438 (1968).

20. Fromm, J. E., "Practical Investigation of Convective Difference Approximations of Reduced Dispersion", Phys. Fluids Supplement II, 12, 3-12 (1969).

21. Gentry, R. A., R. E. Martin, and B. J. Daly, "An Eulerian Differencing Method for Unsteady Compressible Flow Problems", J. Comp. Phys., 1, 87-118 (1966).

22. Ghia, K. N., T. P. Torda, and Z. Lavan, "Laminar Mixing of Heterogeneous Axisymmetric Coaxial Confined Jets", Ill. Inst. Tech., Cont. NSG-694, NASA CR-72480 (1968).

23. Gordon, S., and B. J. McBride, "Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations", NASA SP-273 (1971), 245 pages.

24. Harlow, F. H., "Numerical Methods for Fluid Dynamics, An Annotated Bibliography", Los Alamos Scientific Laboratory Report No. LA-4281, 10 pages (1969).

25. Harlow, F. H., and A. A. Amsden, "A Numerical Fluid Dynamics Calculation Method for All Flow Speeds", J. Comp. Phys., 8, 197-213 (1971), see also Los Alamos Scientific Laboratory Report No. LA-DC 12190, 31 pages (1968).

26. Harlow, F. H. and A. A. Amsden, Fluid Dynamics, Los Alamos Scientific Laboratory Monograph, LA-4700 (1971).

27. Harlow, F. H., and A. A. Amsden, "Numerical Calculation of Almost Incompressible Flow", J. Comp. Phys., 3, 80-93 (1968).

28. Harlow, F. H., and B. D. Meixner, "Rise Through the Atmosphere of a Hot Bubble", Los Alamos Scientific Laboratory Report No. LAMS-2770 (1962).

29. Hirt, C. W., "Heuristic Stability Theory for Finite-Difference Equations", J. Comp. Phys., 2, 339-355 (1968).

30. Hirt, C. W., and J. L. Cook, "Calculating Three-dimensional Flows Around Structures and Over Rough Terrain", Los Alamos Scientific Laboratory Report No. LA-DC 13289, 35 pages (1971).

31. Hirt, C. W. and F. H. Harlow, "A General Corrective Procedure for the Numerical Solution of Initial-Value Problems", J. Comp. Phys., 2, 114-119 (1967).

32. Hornbeck, R. W., "Laminar Flow in the Entrance Region of a Pipe", Appl. Sci. Res., 13A, 224-232 (1963).

33. Kannenberg, R. G., "Cinematic-Fortran Subprograms for Automatic Computer Microfilm Plotting", NASA TM-X-1866 (1969) 49 pages.

34. Karplus, W. J., "An Electric Circuit Theory Approach to Finite Difference Stability", Trans. AIEE, 77, pt. I, 210-213 (1958).

35. Kurzrock, J. W., and R. E. Mates, "Exact Numerical Solution of the Time-Dependent Compressible Navier-Stokes Equations", AIAA 3rd Aerospace Sci. Mtg., New York, N.Y., Jan. 24-26, 1966, paper no. 66-30, 54 pages.

36. Lawson, C., N. Block, and R. Garrett, "Fortran IV Subroutines for Contour Plotting", Jet Propulsion Laboratory Tech. Memo. No. 106 (1966), 53 pages.

37. Lax, P. D., "Hyperbolic Difference Equations: A Review of the Courant-Friedrichs-Lewy Paper in the Light of Recent Developments", IBM Journal, 11, 235-238 (1967).

38. Lax, P., and B. Wendroff, "Systems of Conservation Laws", Comm. Pure and App. Math., 13, 217-237 (1960).

39. MacCormack, R. W., "The Effect of Viscosity in Hypervelocity Impact Cratering", AIAA Hypervelocity Impact Conf., Cincinnati, Ohio, April 30 - May 2, 1969, paper no. 69-354, 8 pages.

40. O'Brien, G. G., M. A. Hyman, and S. Kaplan, "A Study of the Numerical Solution of Partial Differential Equations", J. Math. and Phys., 29, 223-251 (1950).

41. Paris, J., and S. Whitaker, "Confined Wakes: A Numerical Solution of the Navier-Stokes Equations", AIChE J., 11, 1033-1041 (1965).

42. Phillips, B. M., "A Numerical Investigation of Laminar Newtonian Flow in a Sudden Expansion and Tee", PhD. thesis, Dept. of Chem. Eng., Univ. of Tennessee, 177 pages (1968).

43. Rich, M., "A Method for Eulerian Fluid Dynamics", Los Alamos Scientific Laboratory Report No. LAMS-2826, 97 pages (1963).

44. Richtmyer, R. D. and K. W. Morton, Difference Methods for Initial-Value Problems, 2nd Edition, Interscience Publishers, New York, London, Sidney (1967).

45. Roache, P. J., Computational Fluid Dynamics, Hermosa Publishers, Albuquerque, New Mexico (1972).

46. Schwab, T. H., "Steady Free Convection in a Closed Vertical Annular Container", M.S. thesis, Dept. of Chem. Eng., Univ. of Toledo, 165 pages (1968).

47. Seider, W. D., "Confined Jet Mixing in the Entrance Region of a Tubular Reactor", Ph.D. Thesis, Dept. of Chemical Engineering, University of Michigan (1966) 136 pages. See also, Seider, W. D., and S. W. Churchill, "Confined Jet Mixing in the Entrance Region of a Tubular Reactor", AIChE J., 17, 704-712 (1971).

48. Shapiro, A. H., "The Dynamics and Thermodynamics of Compressible Fluid Flow", Vol. I, pp. 91-94, Ronald Press, New York (1953).

49. Svehla, R. A., Personal Communication (1972).

50. Uchida, S., "The Pulsating Viscous Flow Superposed on the Steady Laminar Motion of Incompressible Fluid in a Circular Pipe". ZAMP, 8, 403-422 (1956).

51. Von Neumann, J., and R. D. Richtmyer, "A Method for the Numerical Calculation of Hydrodynamic Shocks", J. App. Phys., 21, 232-237 (1950).

52. Von Rosenberg, D. U., Methods for the Numerical Solution of Partial Differential Equations, Elsevier, New York (1969).

53. Warpinski, N. R., H. M. Nagib, and Z. Lavan, "Experimental Investigation of Recirculating Cells in Laminar Coaxial Jets", AIAA 10th Aerospace Sci. Mtg., San Diego, Calif., Jan. 17-19, 1972, paper no. 72-150, 9 pages.

54. Weinstein, H., and C. A. Todd, "Analysis of Mixing of Coaxial Streams of Dissimilar Fluids Including Energy-Generation Terms", NASA TN D-2123 (1964) 57 pages.

55. Welch, J. E., F. H. Harlow, J. P. Shannon, and B. J. Daly, "The MAC Method, A Computing Technique for Solving Viscous, Incompressible, Transient Fluid-Flow Problems Involving Free Surfaces", Los Alamos Scientific Laboratory Report No. LA 3425 (1966).

56. Williams, F. A., "Theory of Combustion in Laminar Flows", in Annual Review of Fluid Mechanics, Vol. 3, edited by Van Dyke, M., W. G. Vincenti, and J. V. Wehausen, Annual Reviews Inc., Palo Alto, Calif. (1971) pp. 171-188.